Perpustakaan SKTM

NG YIN CHERN
WEK000018

INTEGRATED INFORMATION SYSTEM VIA WIFI TECHNOLOGY

WXES 3181: LAPORAN PROJEK ILMIAH TAHAP AKHIR

Faculty of Computer Science and Information Technology
2002/2003

University of Malaya

Table of Contents

University of Malaya

University of Malaya

University of Malaya

University of Malaya

Table of Figures

University of Malaya

Table of Tables

University of Malaya

Acknowledgement

First and foremost, I would like to acknowledge my supervisor and moderator of this project, namely Mr. Ang Tan Fong and Dr. Rosli Salleh. Mr. Ang has tremendously helped in the process of the planning and working of this documentation of the project. I would also like to thank him especially for taking us under his supervision. I would also like to give a special thanks to Mr. Ling for his views and advices during the weekly presentation of our system design and documentations. His views have been very conducive and constructive. Also, I would like to thank the staffs of the Faculty of Computer Science and Information Technology for preparing the guidelines and materials for this documentation.

Secondly, I would like to acknowledge my partner for this project, Alvin Alan Ilangovan. His help has been tremendously and I would doubt that I would finish this documentation without his help and patience.

I would also like to thank the various contacts and friends which have helped me in the process of research and design of this project. They have helped me in understanding the technology and development process involved in this project.

Lastly, a special thanks to the webmaster of several online resources which I have used for research for this project. Without them, there would not be a complete and useful resource on the Internet.

University of Malaya

Abstract

In the future workspace, the ability to make decision based on live and updated data is ultimately important. The key motivation of this project therefore to enable and enhance data mobility. Wireless LAN (WiFi) application is becoming more popular and there are a growing number of WiFi hotspots around the country. This makes it feasible and suitable for the project to utilize the WiFi technology as its core concept. Also, portable devices such as the PocketPCs are now powerful enough for a full blown application. The project, namely the Integrated Information System via WiFi Technology sought to solve are enhancing data mobility and also enhance data integration by utilizing both the groundbreaking and new WiFi and PDA technologies.

The outcome of this project is to create a fully synchronize-able information system environment using wireless technology, updating information and schedules to the users of this system even when they are on the move securely via its strong security encryption which is DES/TDES.

A very modular approach in the development of the project is being taken. By being modular, the codes could easily be reusable throughout the system. This would easily solve the important issue of data integration. This project would try to solve this by being able to utilize and connect to different types of database. This could be done as the development of the project is very modular. Therefore, the groundbreaking .NET Framework is the main development tool that would be used in the development of this project. Also, MS SQL Server 2000, MS Access and lastly, PocketAccess databases would be used depending on the different platforms in the systems.

University of Malaya

Chapter 1

1.1 Overview

With K-economy prevailing in our country, data and information is getting more important everyday. Although the need for fast access to data has become even more relevant in these days of high speed internet access and high speed transmission mediums such as the broad band and ISDN lines available, but without being able to have relevant data at our hands at every time, the technology is quite irrelevant. Statically sitting down in front of your desktop and logging onto certain network and therefore into your database is not the working environment in the future workspace. This is where the need for a wireless application comes in. Hence I decided to do a project based on the title "Integrated Information Systems via WIFI technology."

As stated, in the future workspace there is a strong need for up-to-date data to work effectively. Schedules and appointments are quite useless if they are out of date. Some might wonder that a mobile device such as a mobile phone would be good enough to receive this information through SMS or even MMS! This actually is a novel idea. However, A hand-phone only partially answers this question as SMS's are considered too short in the real life working environment as much of the data that people deal with is much bigger then the 350 words allowed to be transmitted via one SMS. Therefore, the use of PDA's in a wireless LAN is the total solution for this problem. In a wireless LAN a user is allowed to transmit the amount of data just as though he was in a wired LAN. This time, the user has the advantage of being

University of Malaya

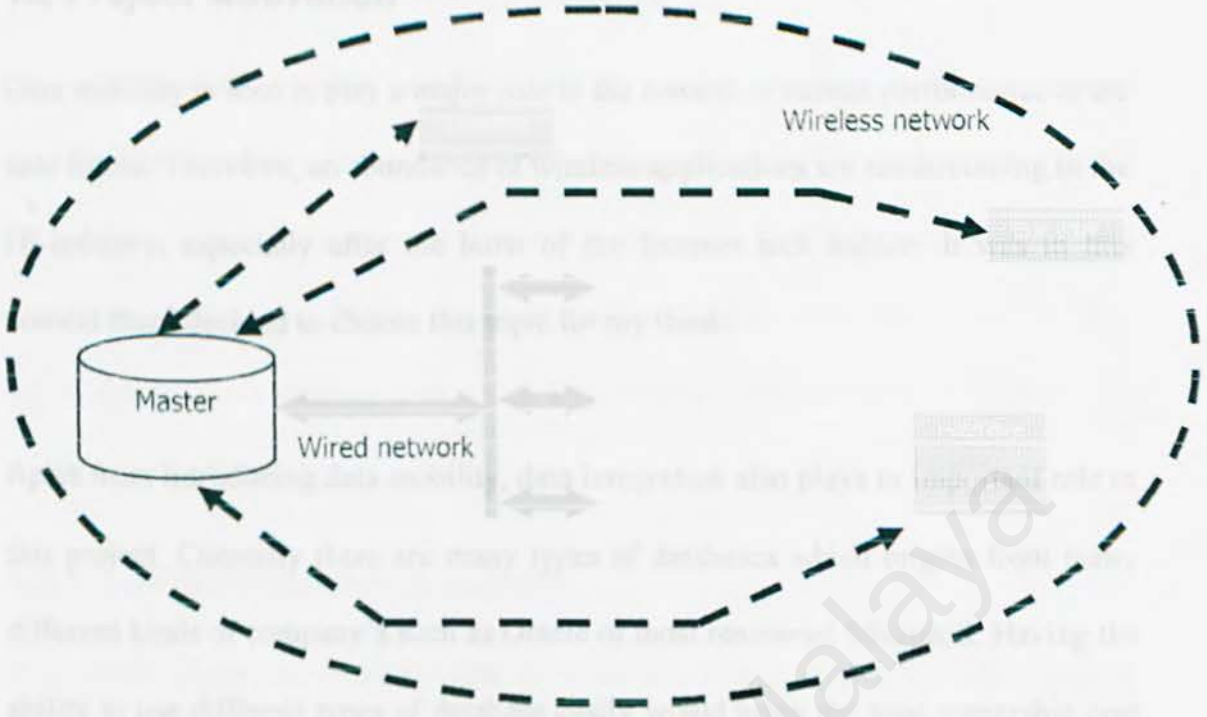


Figure 1: Physical Layout of the system

Figure 2: Logical Layout of the System

University of Malaya

1.2 Project Motivation

Data mobility is seen to play a major role in the context of human performance in the near future. Therefore, an abundance of wireless applications are mushrooming in the IT industry, especially after the burst of the Internet tech bubble. It was in this context that I decided to choose this topic for my thesis.

Apart from introducing data mobility, data integration also plays in important role in this project. Currently there are many types of databases which origins from many different kinds of company's such as Oracle or most renowned Microsoft. Having the ability to use different types of database easily would bring the total ownership cost (TOC) of this system at a bare minimum. In fact, this issue is an on going challenge to most system developers as they decide on the types of database to use for their system.

1.2.1 Wired Applications

Wired applications are what are currently used by most organizations. This means that a user's computer or lap top has to be connected via cables into a network hub or switch in order to access the data that is available. This is particularly concerned when a user is operating in a distributed system environment. Using a distributed system allows a user to have many benefits such as the reduction of workloads, more space to allocate data and faster programming techniques.

University of Malaya

1.2.2 Problems with Wired Applications

The most obvious problem would be that in a wired application the user is not mobile. The user has a limited range of mobility; which is the maximum length of the cables or wires. This situation does not permit users free move around while still manage to work on his device. Being so restricts them to work only whenever they are in front of their desktop computers. This results in the reduction of productivity in any organization.

1.2.3 How this problem is solved

It is rather important to enable data mobility in an integrated information system. With data mobility work is no more limited to a single static node but basically anywhere within the wireless LAN hotspot. Work may even be conducted at off-working hour, like during lunch time in the canteen where wireless LAN is available. Users that have mobile devices such as the PocketPC PDAs would definitely benefit in such a working environment. They could to attend to important activities although they are not in their office. Such is the advantage of wireless applications. By being able to access relevant data at any time and any place, work efficiency and productivity is improved.

This project demonstrates the classic use of latest technology to enhance working environment.

University of Malaya

1.3 Project Objective

The objective of this thesis is to create a fully synchronize-able information system environment using wireless technology. This system would allow up to date information and schedules to the users of this system even when they are on the move. It also enable and enhances data mobility within an organization be it a company or a university. Vital information would be able to be presented to the required person when necessary. It also manages to keep the data transfer from the host to the client secure via its strong security encryption which is DES/TDES.

University of Malaya

1.4 Project Scope

The scope of this thesis consists of a various number of relatively new technologies such as the WiFi aka 802.11b Wireless LAN. Wifi refers to the new standard in which the IEEE has devised to synchronize usage of all wireless applications. More of this standard would be elaborated in chapter 2.

This project also uses the PocketPC or PDAs as means of its mobile device. The PDA is a handheld device which is gaining rapid popularity among the public not least because of its compactness and size. It is a state of the art device which was created for the sole purpose of giving frequent users of PC mobility without the hassle of carrying an object that is too large.

The thing that creates the most hype in this thesis however would have to be the utilization of the .NET Framework. This is the latest development kit to come out of Microsofts team of wizards. The .NET framework features many new key features that make programming simpler yet maintaining its high standards of programmability. More on the .NET framework would be found on Chapter 2.

This project will also see the utilization of the MS SQL Server 2000 Database. AS it is the most common database being used it was only practical for me to use this database and thus enhancing my knowledge on this product.

University of Malaya

1.5 Expected Outcome

This thesis is expected to create a wireless integration system that is capable of synchronizing relevant data such as:

- Schedules / Appointments
- Contacts
- Memos / Notes
- A offline custom form synchronization module

Utilizing a modular development approach, the network module would be highly adaptable where new features and modules can easily interface with it. Other possible modules and features are as the following:-

- Real-time chat/messaging module
- Direct data/file transfer between clients
- Encryption of data base on user's key
- General Bulletin Board

University of Malaya

1.6 Work Schedule

Project Stage	June				July				Aug				Sep				Oct				Nov				Dec				Jan
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1
Preliminary Study and Planning																													
Literature Study																													
System Analysis																													
System Design																													
Prototype																													
Development and Coding																													
Unit Testing																													
System Testing																													
Documentation																													
Implementation and Maintenance																													

Figure 3: Gantt chart of the work schedule

University of Malaya

Chapter 2: Review of Literature

2.1 Analysis Studies

2.1.1 Case Study 1 – Wireless LAN Campus at Carnegie Mellon University

The purpose of this case study is to present an understanding of how a wireless LAN research project located at Carnegie Mellon University was developed and how this technology works. It is also an overview on how a wireless campus operates as this bears many similarities with my thesis. Over the past several years the technology used in wireless LANs has grown. The research team at Carnegie Mellon University was one of the leading innovators in this development. The innovation and hard work of the research team along with Lucent Technology has developed a state of the art wireless LAN called "Wireless Andrew."

In 1994 Carnegie Mellon University (CMU) submitted a request to the National Science Foundation (NSF) to create a High Speed Wireless LAN infrastructure. The plan was to integrate the existing campus network with a wireless LAN so that students and professors with portable computers could maintain consistent connections to the campus-wide computer network known as "Wireless Andrew." Today Carnegie Mellon University, located in Pittsburgh, is applying both CDPD (cellular digital packet data) and Lucent's WaveLAN technology to provide connections for wireless roaming for portable systems.

University of Malaya

The focus of this case study is to present an understanding of the technology used in the design of "Wireless Andrew." In addition the terms and definitions will facilitate the understanding of the underlying concepts covered in this presentation.

The specifications used in determining the vendor for the "Wireless Andrew" project was based on (1) resources, expertise, commitment, (2) a strategy for upgrading the network, (3) a commitment to jointly pursue research. Lucent's span on the wireless industry included practically everything in the wireless LAN industry [1]. After much research and testing the Carnegie Mellon research team decided to use Lucent Technology's WaveLAN® products to construct "Wireless Andrew."

2.1.1.1 WaveLAN® Technology

Lucent's WaveLAN® technology uses a radio transmission technique known as DSSS (Direct Sequence Spread Spectrum). This transmission technique is used in spread spectrum radio-transmission systems. Spread Spectrum is most commonly used in various wireless LAN technologies and PCS cellular systems. We will first examine the WaveLAN technology before understanding how "Wireless Andrew" works.

2.1.1.2 Peer-to-Peer WaveLAN®

All that is needed for two computers to communicate over a Peer-to-Peer wireless LAN is two computers with wireless LAN adapter cards.

University of Malaya

High speed wireless LANs is no longer limited to the building infrastructure. The wireless LAN interface cards use radio frequencies to communicate with each other, allowing the user to transfer data without having a personal computer or mobile computer physically attached to a wired LAN or telephone line. Any two computers that are within range can set up independent communications. The standard range distance for a wireless PC adapter card is 1200 ft in an open environment and 550 feet in a semi-open environment. The WaveLAN® interface cards come in PCMIA, ISA, and EISA (see Figure2. WaveLAN PC Card).

To extend a wired LAN you will need to install a WavePOINT ® wireless access point to your wired LAN. WavePOINT ® Access Points add wireless workstations and other mobile devices to the wired network (see Figure 4: Extended Wired LAN). The access point acts as a repeater or bridge effectively doubling the distance between computers. The multiple access points in the diagram are the small boxes connected directly to the Ethernet Backbone. The WaveLAN® access points allow users to share resources by linking the roaming computers to the wired part of the network.

Figure 4: Extended Wired LAN

2.1.1.3 Components of "Wireless Andrew"

The technology used in the "Wireless Andrew" project contains only a few more components than what you would normally find in a Peer-to-Peer wireless LAN. The architecture of the project fits the extended wired LAN concept (see Figure 5: "Wireless Andrew" Core Wired LAN Components).

For the "Wireless Andrew" project the PC Card used in laptop computers is the WaveLAN® PCMIA wireless LAN interface card along with the Cellular Digital Packet modem s (CDPD). The CDPD service permits roaming outside the campus network throughout the Pittsburgh area. CDPD also supports Internet Protocols (IP) that allows the CDPD network to be linked with the WaveLAN® network.

University of Malaya

CDPD uses the idle channels in the cellular system to provide the connectionless digital packet service (CDPD). The coverage area for the CDPD system has a radius of 1-10 miles. When added to the WaveLAN® network infrastructure it provides additional roaming capability for the campus wide network.

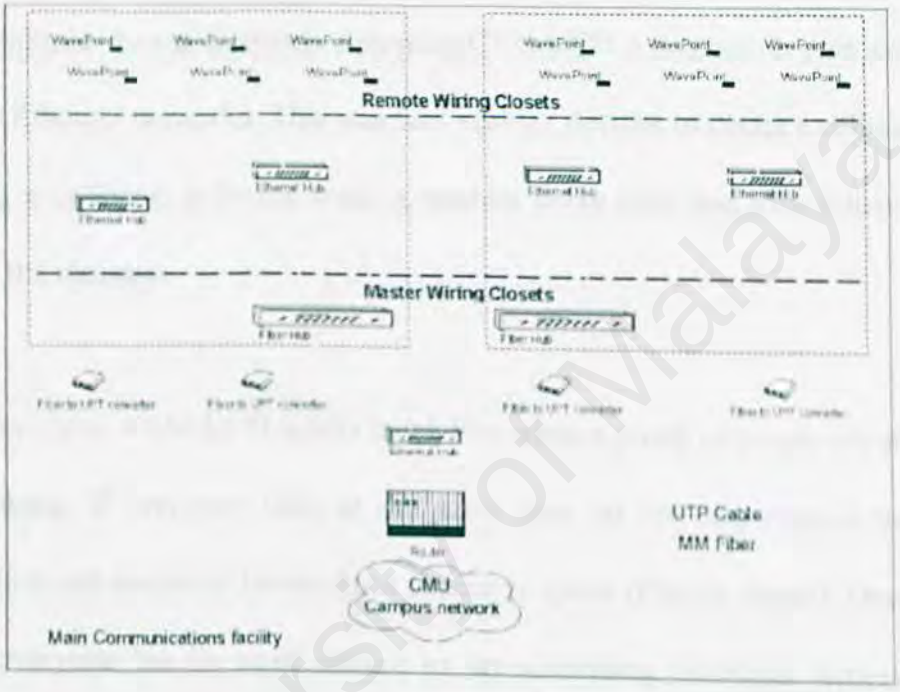


Figure 5: "Wireless Andrew" Core Wired LAN Components

The WaveLAN® technology used in the "Wireless Andrew" project is achieved through the use of a Spread Spectrum modulation technology called direct sequence spread spectrum or DSSS. DSSS operates in an unlicensed radio range at 2.4 Ghz and 915 MHz. Direct-sequence spread-spectrum (DSSS) generates a redundant bit pattern for each bit to be transmitted.

This bit pattern is called a chip (or chipping code). The longer the chip, the greater the probability that the original data can be recovered (and, of course, the more

bandwidth required). Even if one or more bits in the chip are damaged during transmission, statistical techniques embedded in the radio can recover the original data without the need for retransmission.

The data transmission used in the WaveLAN® technology incorporates Carrier Sense Multiple Access/Collision Selection CSMA/CD a contention protocol used in standard Ethernet networks. This standard enables devices to detect a collision. After detecting a collision, a device waits a random delay time and then attempts to re-transmit the message.

In simpler terms, CSMA/CD works much like when a group of people are sitting in a room talking. If everyone talks at the same time no one can receive the correct information and everyone listens for a chance to speak (Carrier Sense). Once a space occurs, everyone has an equal chance to say something (Multiple Access). If two people start talking at the same instant, they detect that fact, and quit speaking (Collision Detection).

The WaveLAN modem prepends a 16-bit "network ID" to every data packet on transmit and can be set to reject all but one network ID on receive. "The "network ID" provides multiple logical Ethernet address spaces, which allows the WaveLAN-to-Ethernet bridges to use standard bridge routing protocol". The WaveLAN access point handles the roaming from cell to cell.

University of Malaya

2.1.1.4 Wireless Roaming

The "Wireless Andrew" network LAN communication is limited to how far the signal can carry. The WaveLAN® technology uses micro cells to extend the range of the wireless connectivity for roaming computers. At any location, a mobile PC with a WaveLAN® adapter is associated with an individual WaveLAN® access point and its associated micro cell. The micro cells overlap to allow continuous coverage for the communication within the wireless LAN. The mobile PCs are handed off as they travel within the micro cell ranges to another access point

2.1.1.5 Structure of Wireless Andrew

Wireless Andrew uses a new IEEE 802.3 backbone network on-campus. It uses this backbone to connect the WavePOINT units in each building (through a separate router located in the main campus data communications facility) with the rest of the campus wired network. In each building, each WavePOINT is connected to a SynOptics IEEE 802.3 10BaseT hub located either in the buildings master wiring closet or in a remote wiring closet on the floor of the building. These hubs, in turn are connected by multimode fiber cable to a 10BaseT hub which is central to the high-speed wireless network. This hub connects to a Cisco router. The WavePOINT units and hubs are powered by the campus 110 VAC system.

This structure allows us to operate the high-speed wireless network independent of the campus network. We will use this arrangement until we are absolutely sure that the wireless network is stable. This structure also allows us to separate traffic on the

University of Malaya

wireless LAN backbone network from traffic on the rest of the network. The Cisco router connecting the wireless backbone to the rest of the campus filters packets based on destination address and only passes packets to and from the wireless backbone as needed. For example, this allows us to avoid passing the packets used for the WaveAROUND sign on protocol between the mobile computer's old and new WavePOINT access point over the main campus network backbone.

2.1.1.6 Result of Study

Strength

1. **Mobility**: Wireless Andrew systems can provide LAN users with access to real-time information anywhere in their organization. This mobility supports productivity and service opportunities not possible with wired networks.
2. **Installation Flexibility**: Wireless technology allows the network to go where wired applications cannot go.
3. **Simplicity in accessing campus data: Wireless** Andrew allows students and staff to access relevant data no matter where they are. For the students they may access their lecture notes via the network and the staff gain access to the relevant data regardless where their location is.

Weaknesses

Wireless Andrew fails to address the common problems associated with wireless LAN which is:

1. **Hidden Stations**: Carrier sensing used in the Wireless Andrew may sometimes fail to detect that another station is transmitting This could lead to the problem of collision within the network.

University of Malaya

2. **Fading**: Due to the inverse square law of electromagnetic wave propagation the strength of radio signals diminish rapidly within the distance from the transmitting stations

University of Malaya

2.1.2 Case Study -Wi-Fi Wireless LAN Connectivity in CompactFlash Format for Pocket PC PDAs (Symbol Technologys)

The Wireless Networker from Symbol Technologies brings the power of high-speed Wi-Fi wireless connectivity to today's popular personal data appliances. It supports PDAs running Microsoft® Windows™ for Pocket PC, fitting devices with CompactFlash Type I/II extended card slots including the Compaq iPAQ, HP Jornada, and Casio Cassiopeia. It also supports notebook computers running Windows 95, 98, ME, or 2000 with a Type II PC Card (PCMCIA) slot via a CF-to-PC Card adapter.

With the Wireless Networker, applications and communications are mobilized at Ethernet-speed, providing cable-free access to corporate and public networks and business-critical email, the Internet, intranets, CRM, sales force automation, and other applications.

2.1.2.1 Strengths of the Wireless Networker

The Wireless Networker's high-performance features provide secure mobility:

- *Advanced Power Management and Suspend on WLAN* ensure low power consumption and prevent against accidental battery drain, extending the life of the PDA battery
- Wi-Fi certification provides multi-vendor compatibility with Wi-Fi certified 802.11b wireless LANs, providing wireless access at 802.11b office, home,

University of Malaya

and public WLAN spaces (optimum performance is realized in a Symbol Spectrum24 High Rate 11 Mbps Wireless LAN environment)

- Data rate up to 11 Mbps supports bandwidth-demanding applications
- Automatic rate scaling at 11, 5.5, 2 and 1 Mbps for maximum range and optimum data throughput
- 40 and 128-bit WEP encryption and decryption provide secure communications
- User-friendly tools make installation and management easy
- A range of 300 ft./91 m ensures extended wireless mobility as you travel through the office or other networked space

2.1.2.2 Uses for this system

The Wireless Networker enables mobile workers in healthcare, education, retail, manufacturing, and hospitality; associates at corporate and regional offices; Pocket PC enthusiasts with a home wireless LAN solution — anyone with access to 802.11b Wi-Fi-certified connectivity — to enjoy increased productivity and improved communications with the power of high-performance wireless mobility. And it's part of Symbol's award-winning Spectrum24 High Rate family of wireless connectivity and mobile appliance and management solutions.

University of Malaya

2.1.2.3 Features and Benefits of this system

To make things clearer I have placed the features and benefits of the Wireless Networker into the table below:

Table 1: Features and benefits of the Wireless Networker

Features	Benefits
11 Mbps direct sequence (DS) high data rate transmission	Supports bandwidth-demanding applications and high-speed data transmission
Supports both Pocket PC and Windows PC operating systems	Flexible to work with both your PDA and your notebook PC
Wi-Fi certified (IEEE 802.11b compliant)	Interoperable wireless networking based on the high rate standard
CompactFlash form factor Type I/II	Supports a variety of popular computing devices, from handhelds to laptops
Automatic data rate and channel selection of 11, 5.5, 2, and 1 Mbps	Optimized throughput and range for continuous connectivity
Wireless software upgrade	Simplified update and maintenance
Wired Equivalent Privacy (WEP) encryption and decryption support	Powerful data security at 40 and 128 bits
User-friendly client tools	Simplified administration and maintenance
Advanced Power Management and Suspend on WLAN	Very low power consumption delivers extended battery life for client devices

University of Malaya

2.1.2.4 Compatibility of the System

Table 2: Compatibility of the Wireless Networker

Compatibility	
Client devices:	• Pocket PC 2000 Devices: - Casio Cassiopeia E-125 Pocket PC - Compaq iPAQ H 3100 and 3600 Series Pocket PC (requires CF Jacket accessory or PC Card Jacket accessory and a CF-to-PC Card adapter) - HP Jornada 520 and 540 Series Pocket PCs • Pocket PC 2002 Devices • Handheld PC 2000 Devices • Notebook PC with Windows 95/98/ME/ 2000 and a Type II PC Card (PCMCIA) slot (requires a CF-to-PC Card adapter)
Radio Characteristics	
Frequency Range:	U.S., Europe, Japan product covering 2.4 to 2.5 GHz, programmable for different country regulations
Data Rate:	11 Mbps per channel maximum with Dynamic Rate Scaling to 5.5,

University of Malaya

	2 and 1 Mbps
Output Power:	100 mW, programmable for different country regulations
Power Management:	Continuously Aware and Power Saving Modes
Range:	Data throughput scaled to support up to 300 ft./91 m indoor/outdoor
TX Maximum Radiated EIRP:	FCC regulations part 15.247 in U.S.; ETS 300 328 in Europe; RCR STD-33 in Japan
Modulation:	Direct Sequence Spread Spectrum (DSSS) with BPSK (1 Mbps), QPSK (2 Mbps), and CCK (5.5 and 11 Mbps)
TX Out-of-Band Emissions:	FCC regulations part 15.247, 15.205, 15.209 in U.S.; ETS 300 328 in Europe; RCR STD-33 in Japan
Operating Temperature:	-20° to 70° C/-4° to 158° F
Storage Temperature:	-30° to 80° C/-22° to 176° F

Network Characteristics

Driver Software Supported:	Pocket PC (CE 3.0 9348), HPC 2000, Pocket PC 2002, Windows 95/98/ME/2000
Access Protocol:	CSMA/CA
Roaming:	Virtually instantaneous

University of Malaya

2.2 Network

A network is nothing more than two or more computers connected together through a medium for data exchange. Although there many types of network technologies that is considerable for this project; such as WiFi, Bluetooth, InfraRed LAN or even RF LAN, it is actually most feasible to select the IEEE 802.11b. This is simply because of the growing popularity of this standard compared to the others available wireless LAN technology.

2.2.1 Wireless LAN

A wireless local area network (LAN) is a flexible data communications system implemented as an extension to or as an alternative for, a wired LAN. Using radio frequency (RF) technology, wireless LANs transmit and receive data over the air, minimizing the need for wired connections. Thus, wireless LANs combine data connectivity with user mobility.

Wireless LANs have gained strong popularity in a number of vertical markets, including the health-care, retail, manufacturing, warehousing, and academia. These industries have profited from the productivity gains of using hand-held terminals and notebook computers to transmit real-time information to centralized hosts for processing. Today wireless LANs are becoming more widely recognized as a general-purpose connectivity alternative for a broad range of business customers. Business Research Group, a market research firm, predicts a six fold expansion of the worldwide wireless LAN market by the year 2000, reaching more than $2 billion in revenues.

University of Malaya

Advantages of Wireless LAN

The widespread reliance on networking in business and the meteoric growth of the Internet and online services are strong testimonies to the benefits of shared data and shared resources. With wireless LANs, users can access shared information without looking for a place to plug in, and network managers can set up or augment networks without installing or moving wires. Wireless LANs offer the following productivity, convenience, and cost advantages over traditional wired networks:

- **Mobility**: Wireless LAN systems can provide LAN users with access to real-time information anywhere in their organization. This mobility supports productivity and service opportunities not possible with wired networks.
- **Installation Speed and Simplicity**: Installing a wireless LAN system can be fast and easy and can eliminate the need to pull cable through walls and ceilings.
- **Installation Flexibility**: Wireless technology allows the network to go where wire cannot go.
- **Reduced Cost-of-Ownership**: While the initial investment required for wireless LAN hardware can be higher than the cost of wired LAN hardware, overall installation expenses and life-cycle costs can be significantly lower. Long-term cost benefits are greatest in dynamic environments requiring frequent moves and changes.
- **Scalability**: Wireless LAN systems can be configured in a variety of topologies to meet the needs of specific applications and installations. Configurations are easily changed and range from peer-to-peer networks

University of Malaya

suitable for a small number of users to full infrastructure networks of thousands of users that enable roaming over a broad area.

How Wireless LANs Are Used in the Real World

Wireless LANs frequently augment rather than replace wired LAN networks—often providing the final few meters of connectivity between a wired network and the mobile user. The following list describes some of the many applications made possible through the power and flexibility of wireless LANs:

- Doctors and nurses in hospitals are more productive because hand-held or notebook computers with wireless LAN capability deliver patient information instantly.
- Consulting or accounting audit teams or small workgroups increase productivity with quick network setup.
- Students holding class on a campus greensward access the Internet to consult the catalog of the Library of Congress.
- Network managers in dynamic environments minimize the overhead caused by moves, extensions to networks, and other changes with wireless LANs.
- Training sites at corporations and students at universities use wireless connectivity to ease access to information, information exchanges, and learning.
- Network managers installing networked computers in older buildings find that wireless LANs are a cost-effective network infrastructure solution.
- Trade show and branch office workers minimize setup requirements by installing pre-configured wireless LANs needing no local MIS support.

University of Malaya

- Warehouse workers use wireless LANs to exchange information with central databases, thereby increasing productivity.
- Network managers implement wireless LANs to provide backup for mission-critical applications running on wired networks.
- Senior executives in meetings make quicker decisions because they have real-time information at their fingertips.

Wireless LAN Technology

Manufacturers of wireless LANs have a range of technologies to choose from when designing a wireless LAN solution. Each technology comes with its own set of advantages and limitations.

Narrowband Technology

A narrowband radio system transmits and receives user information on a specific radio frequency. Narrowband radio keeps the radio signal frequency as narrow as possible just to pass the information. Undesirable crosstalk between communications channels is avoided by carefully coordinating different users on different channel frequencies.

A private telephone line is much like a radio frequency. When each home in a neighborhood has its own private telephone line, people in one home cannot listen to calls made to other homes. In a radio system, privacy and noninterference are accomplished by the use of separate radio frequencies. The radio receiver filters out all radio signals except the ones on its designated frequency.

University of Malaya

From a customer standpoint, one drawback of narrowband technology is that the end-user must obtain an FCC license for each site where it is employed.

Spread Spectrum Technology

Most wireless LAN systems use spread-spectrum technology, a wideband radio frequency technique developed by the military for use in reliable, secure, mission-critical communications systems. Spread-spectrum is designed to trade off bandwidth efficiency for reliability, integrity, and security. In other words, more bandwidth is consumed than in the case of narrowband transmission, but the tradeoff produces a signal that is, in effect, louder and thus easier to detect, provided that the receiver knows the parameters of the spread-spectrum signal being broadcast. If a receiver is not tuned to the right frequency, a spread-spectrum signal looks like background noise. There are two types of spread spectrum radio: frequency hopping and direct sequence.

Frequency-Hopping Spread Spectrum Technology

Frequency-hopping spread-spectrum (FHSS) uses a narrowband carrier that changes frequency in a pattern known to both transmitter and receiver. Properly synchronized, the net effect is to maintain a single logical channel. To an unintended receiver, FHSS appears to be short-duration impulse noise.

Direct-Sequence Spread Spectrum Technology

Direct-sequence spread-spectrum (DSSS) generates a redundant bit pattern for each bit to be transmitted. This bit pattern is called a chip (or chipping code). The longer

University of Malaya

the chip, the greater the probability that the original data can be recovered (and, of course, the more bandwidth required). Even if one or more bits in the chip are damaged during transmission, statistical techniques embedded in the radio can recover the original data without the need for retransmission. To an unintended receiver, DSSS appears as low-power wideband noise and is rejected (ignored) by most narrowband receivers.

Infrared Technology

A third technology, little used in commercial wireless LANs, is infrared. Infrared (IR) systems use very high frequencies, just below visible light in the electromagnetic spectrum, to carry data. Like light, IR cannot penetrate opaque objects; it is either directed (line-of-sight) or diffuse technology. Inexpensive directed systems provide very limited range (3 ft) and typically are used for personal area networks but occasionally are used in specific wireless LAN applications. High performance directed IR is impractical for mobile users and is therefore used only to implement fixed sub-networks. Diffuse (or reflective) IR wireless LAN systems do not require line-of-sight, but cells are limited to individual rooms.

2.2.1.4 How Wireless LANs Work

Wireless LANs use electromagnetic airwaves (radio or infrared) to communicate information from one point to another without relying on any physical connection. Radio waves are often referred to as radio carriers because they simply perform the function of delivering energy to a remote receiver. The data being transmitted is superimposed on the radio carrier so that it can be accurately extracted at the

University of Malaya

receiving end. This is generally referred to as modulation of the carrier by the information being transmitted. Once data is superimposed (modulated) onto the radio carrier, the radio signal occupies more than a single frequency, since the frequency or bit rate of the modulating information adds to the carrier.

Multiple radio carriers can exist in the same space at the same time without interfering with each other if the radio waves are transmitted on different radio frequencies. To extract data, a radio receiver tunes in one radio frequency while rejecting all other frequencies.

In a typical wireless LAN configuration, a transmitter/receiver (transceiver) device, called an access point, connects to the wired network from a fixed location using standard cabling. At a minimum, the access point receives, buffers, and transmits data between the wireless LAN and the wired network infrastructure. A single access point can support a small group of users and can function within a range of less than one hundred to several hundred feet. The access point (or the antenna attached to the access point) is usually mounted high but may be mounted essentially anywhere that is practical as long as the desired radio coverage is obtained.

End users access the wireless LAN through wireless-LAN adapters, which are implemented as PC cards in notebook or palmtop computers, as cards in desktop computers, or integrated within hand-held computers. Wireless LAN adapters provide an interface between the client network operating system (NOS) and the

University of Malaya

airwaves via an antenna. The nature of the wireless connection is transparent to the NOS.

2.2.1.5 Wireless LAN Configurations

Wireless LANs can be simple or complex. At its most basic, two PCs equipped with wireless adapter cards can set up an independent network whenever they are within range of one another. This is called a peer-to-peer network. On-demand networks such as in this example require no administration or pre-configuration. In this case each client would only have access to the resources of the other client and not to a central server.

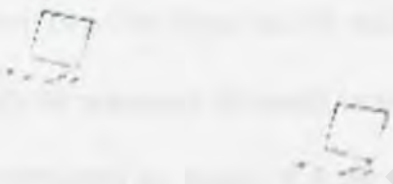


Figure 6: A wireless peer-to-peer network

Installing an access point can extend the range of an ad hoc network, effectively doubling the range at which the devices can communicate. Since the access point is connected to the wired network each client would have access to server resources as well as to other clients. Each access point can accommodate many clients; the specific number depends on the number and nature of the transmissions involved. Many real-world applications exist where a single access point services from 15-50 client devices.

University of Malaya

Figure 7: Client and Access Point

Access points have a finite range, on the order of 500 feet indoor and 1000 feet outdoors. In a very large facility such as a warehouse, or on a college campus it will probably be necessary to install more than one access point. Access point positioning is accomplished by means of a site survey. The goal is to blanket the coverage area with overlapping coverage cells so that clients might range throughout the area without ever losing network contact. The ability of clients to move seamlessly among a cluster of access points is called *roaming*. Access points hand the client off from one to another in a way that is invisible to the client, ensuring unbroken connectivity.

Figure 8: Multiple access points and roaming

University of Malaya

To solve particular problems of topology, the network designer might choose to use Extension Points to augment the network of access points. Extension Points look and function like access points, but they are not tethered to the wired network as are APs. EPs function just as their name implies: they extend the range of the network by relaying signals from a client to an AP or another EP. EPs may be strung together in order to pass along messaging from an AP to far-flung clients, just as humans in a bucket brigade pass pails of water hand-to-hand from a water source to a fire.

Figure 9: Use of an extension point

One last item of wireless LAN equipment to consider is the directional antenna. Let's suppose you had a wireless LAN in your building A and wanted to extend it to a leased building, B, one mile away. One solution might be to install a directional antenna on each building, each antenna targeting the other. The antenna on A is connected to your wired network via an access point. The antenna on B is similarly connected to an access point in that building, which enables wireless LAN connectivity in that facility.

University of Malaya

Figure 10: The use of directional antennas

2.2.1.6 Considerations when implementing a wireless network

While wireless LANs provides installation and configuration flexibility and the freedom inherent in network mobility, customers should be aware of the following factors when considering wireless LAN systems.

Range and coverage

The distance over which RF and IR waves can communicate is a function of product design (including transmitted power and receiver design) and the propagation path, especially in indoor environments. Interactions with typical building objects, including walls, metal, and even people, can affect how energy propagates, and thus what range and coverage a particular system achieves. Solid objects block infrared signals, which impose additional limitations. Most wireless LAN systems use RF because radio waves can penetrate most indoor walls and obstacles. The range (or radius of coverage) for typical wireless LAN systems varies from under 100 feet to more than 300 feet. Coverage can be extended and true freedom of mobility via roaming, provided through micro cells.

University of Malaya

Throughput

As with wired LAN systems, actual throughput in wireless LANs is product and set-up dependent. Factors that affect throughput include the number of users, propagation factors such as range and multi paths, the type of wireless LAN system used, as well as the latency and bottlenecks on the wired portions of the LAN. Data rates for the most widespread commercial wireless LANs are in the 1.6 Mbps range. Users of traditional Ethernet or Token Ring LANs generally experience little difference in performance when using a wireless LAN. Wireless LANs provide throughput sufficient for the most common LAN-based office applications, including electronic mail exchange, access to shared peripherals, Internet access, and access to multi-user databases and applications.

As a point of comparison, it is worth noting that state-of-the-art V.90 modems transmit and receive at optimal data rates of 56.6 Kbps. In terms of throughput, a wireless LAN operating at 1.6 Mbps is almost thirty times faster. However, as the Wireless LAN technology improves and matures, greater speeds can be achieved. As of now, the 802.11b specifications allows up till 11mbps and even new specification for higher speeds (22mbps) are nearing availability.

Integrity and Reliability

Wireless data technologies have been proven through more than fifty years of wireless application in both commercial and military systems. While radio interference can cause degradation in throughput, such interference is rare in the workplace. Robust designs of proven wireless LAN technology and the limited

University of Malaya

distance over which signals travel result in connections that are far more robust than cellular phone connections and provide data integrity performance equal to or better than wired networking.

Compatibility with the Existing Network

Most wireless LANs provide for industry-standard interconnection with wired networks such as Ethernet or Token Ring. Wireless LAN nodes are supported by network operating systems in the same fashion as any other LAN node: thought the use of the appropriate drivers. Once installed, the network treats wireless nodes like any other network component.

Interoperability of Wireless Devices

Customers should be aware that wireless LAN systems from different vendors might not be interoperable for three reasons. First, different technologies will not interoperate. A system based on spread spectrum frequency hopping (FHSS) technology will not communicate with another based on spread spectrum direct sequence (DSSS) technology. Second, systems using different frequency bands will not interoperate even if they both employ the same technology. Third, systems from different vendors may not interoperate even if they both employ the same technology and the same frequency band, due to differences in implementation by each vendor.

Interference and Coexistence

The unlicensed nature of radio-based wireless LANs means that other products that transmit energy in the same frequency spectrum can potentially provide some

University of Malaya

measure of interference to a wireless LAN system. Microwave ovens are a potential concern, but most wireless LAN manufacturers design their products to account for microwave interference. Another concern is the co-location of multiple wireless LANs. While wireless LANs from some manufacturers interfere with wireless LANs, others coexist without interference. This issue is best addressed directly with the appropriate vendors.

Simplicity/Ease of Use

Users need very little new information to take advantage of wireless LANs. Because the wireless nature of a wireless LAN is transparent to a user's NOS, applications work the same as they do on wired LANs. Wireless LAN products incorporate a variety of diagnostic tools to address issues associated with the wireless elements of the system; however, products are designed so that most users rarely need these tools.

Wireless LANs simplify many of the installation and configuration issues that plague network managers. Since only the access points of wireless LANs require cabling, network managers are freed from pulling cables for wireless LAN end users. Lack of cabling also makes moves, adds, and changes trivial operations on wireless LANs. Finally, the portable nature of wireless LANs lets network managers preconfigure and troubleshoot entire networks before installing them at remote locations. Once configured properly wireless LANs can be moved from place to place with little or no modification.

University of Malaya

Security

Because wireless technology has roots in military applications, security has long been a design criterion for wireless devices. Security provisions are typically built into wireless LANs, making them more secure than most wired LANs. It is extremely difficult for unintended receivers (eavesdroppers) to listen in on wireless LAN traffic. Complex encryption techniques make it impossible for all but the most sophisticated to gain unauthorized access to network traffic. In general, individual nodes must be security-enabled before they are allowed to participate in network traffic.

Cost

A wireless LAN implementation includes both infrastructure costs, for the wireless access points, and user costs, for the wireless LAN adapters. Infrastructure costs depend primarily on the number of access points deployed; access points range in price from $1,000 to $2000. The number of access points typically depends on the required coverage region and/or the number and type of users to be serviced. The coverage area is proportional to the square of the product range. Wireless LAN adapters are required for standard computer platforms, and range in price from $300 to $1,000.

The cost of installing and maintaining a wireless LAN generally is lower than the cost of installing and maintaining a traditional wired LAN, for two reasons. First, a wireless LAN eliminates the direct costs of cabling and the labor associated with installing and repairing it. Secondly, wireless LANs simplifies movement, addition,

University of Malaya

and change, allowing a reduction of indirect costs, such as user downtime and administrative overhead.

Scalability

Wireless networks can be designed to be extremely simple or quite complex. Wireless networks can support large numbers of nodes and/or large physical areas by adding access points to boost or extend coverage.

Battery Life for Mobile Platforms

End-user wireless products are designed to run off the AC or battery power from their host notebook or hand-held computer, since they have no direct wire connectivity of their own. Wireless LAN vendors typically employ special design techniques to maximize the host computer's energy usage and battery life.

Safety

The output power of wireless LAN systems is very low, much less than that of a hand-held cellular phone. Since radio waves fade rapidly over distance, very little exposure to RF energy is provided to those in the area of a wireless LAN system. Wireless LANs must meet stringent government and industry regulations for safety. No adverse health affects have ever been attributed to wireless LANs.

University of Malaya

2.2.1.7 Summary

Flexibility and mobility make wireless LANs both effective extensions and attractive alternatives to wired networks. Wireless LANs provide all the functionality of wired LANs, without the physical constraints of the wire itself. Wireless LAN configurations range from simple peer-to-peer topologies to complex networks offering distributed data connectivity and roaming. Besides offering end-user mobility within a networked environment, wireless LANs enable portable networks, allowing LANs to move with the knowledge workers that use them.

2.2.2 WIFI (IEEE 802.11b)

2.2.2.1 Overview of the standard

The IEEE 802.11 standard extends the carrier sensing multiples access (CSMA) principle employed from Ethernet (IEEE 802.3) technology to suit the characteristics of wireless communication. This standard is intended to support communication between computers located within about 150 meters of one another at speeds up to 11Mbps.

The stations in IEEE 802.11 networks use radio frequency signals (in the 2.4 Ghz band) or infra-red signaling as the transmission medium. The radio version of the standard has received the most commercial attention. It uses various frequency selection and frequency hopping techniques to avoid external interface and mutual interference between independent wireless LANs

University of Malaya

The 802.11 protocol offers equal opportunity to all stations to use the transmission channel and any station may transmit directly to any other

2.2.2.2 Problems with using radio frequencies

Several problems arise with using radio waves rather then wires as the transmission medium. These problems stem from the fact that the carrier sensing and collision detecting mechanisms are effective only when the strength of the signal is approximately the same throughout the network.

Recalling that the purpose of carrier sensing is to determine weather the medium is free at all points between the sending and receiving stations and that of collision detection is to determine weather the medium in the vicinity of the receiver is free from interference during transmission. The failure in wireless LANs because signal strength is not uniform of which it operates is:

- Hidden stations: Carrier sense may fail to detect that another station on the network is transmitting.
- Fading: Due to the inverse square law of electromagnetic wave propagation the strength of radio signals diminishes rapidly with the distance from the transmitting station. Stations in a wireless LAN may be out of range of other stations within a wireless LAN.
- Collision masking: The listening technique used in Ethernet detects collisions is not very effective on radio networks. Referring to the inverse square law above the locally generated signals will always be much stronger then signals

University of Malaya

generated anywhere else and that effectively drowns out the remote transmission

2.2.2.3 Overcoming the problems

Despite its fallibility carrier sensing is not dispensed within IEEE 802.11 networks and is augmented with the addition of a slot reservation mechanism to the MAC protocol. The resulting scheme is called *carrier sensing multiple access with collision avoidance* (CSMA/CA).

When the station is ready to transmit it senses the medium. If it detects no carrier signals it may assume that one of the following conditions is true.

- the medium is available
- an out of range station is in the process of requesting the slot
- an out of range station is using the slot that it had previously reserved

The slot reservation protocol involves the exchange pair of short messages (frames) between the intended sending and receiver. The first is a *request to send* (RTS) frame from the sender to the receiver. The RTS message specifies duration for the slot requested. The receiver replies with a *clear to send* (CTS) frame repeating the duration of the slot. The effect of this exchange is as follows:

- Stations within the range of the sender will pick up the RTS frame and take note of the duration
- Stations within the range of the receiver will pick up the RTS frame and take note of the duration

University of Malaya

As a result all of the stations within the range of both sender and receiver will refrain from transmitting for the duration of the requested slot leaving the channel free for the sender to transmit a data frame of the appropriate length .Finally the successful receipt of the data frame is acknowledged by the receiver to help deal with the problem of external interface with the channel. The slot reservation feature of the MAC protocol helps to avoid collision in these ways.

- The CTS helps to avoid hidden stations and fading problems
- The RTS and CTS frames are short so the risk of collision with them is low. If one is detected or an RTS does not result in a CTS a random back-off period is used as in the Ethernet.
- When the RTS and CTS frames have been correctly exchanged there should be no collisions involving the subsequent data and acknowledgement frames unless the intermittent fading prevented a third party from receiving either of them.

University of Malaya

2.3 Hardware Resources

2.3.1 PDA

PDA (Personal Digital Assistant) is a term for any small mobile hand-held device that provides computing and information storage and retrieval capabilities for personal or business use, often for keeping schedule calendars and address book information handy. The term handheld is a synonym. Many people use the name of one of the popular PDA products as a generic term. These include Hewlett-Packard's Palmtop and 3Com's PalmPilot.

Most PDAs have a small keyboard. Some PDAs have an electronically sensitive pad on which handwriting can be received. Apple's Newton, which has been withdrawn from the market, was the first widely-sold PDA that accepted handwriting. Typical uses include schedule and address book storage and retrieval and note-entering. However, many applications have been written for PDAs. Increasingly, PDAs are combined with telephones and paging systems.

A more modern variation of PDAs offer utilizes the Microsoft Windows operating system called Windows CE. Other products have their own or another operating system.

University of Malaya

Figure 11: Types of PDA's Available in today's market

2.4 Wireless LAN Security

Until recently, wireless local-area network (LAN) products were used primarily in certain vertical markets such as retail, education, and healthcare where mobile users with a need for LAN access were satisfied with data-transfer rates of 2 megabits per second (Mbps) or less. Even though most wireless LANs were extensions of wired LANs, the proprietary nature and slow speeds of wireless LANs forced organizations to manage

Wireless LANs are unique entities. To make wireless LANs more "mainstream," customers pressed vendors to develop a high-speed wireless LAN standard that would encourage interoperability, reduce prices, and provide the bandwidth needed by today's business applications. There was also a major concern of the security in using a wires LAN.

The privacy and integrity of communication is an obvious concern for wireless networks. Any station that is within range and is equipped with a transmitter/receiver might seek to join the network or failing that it might eavesdrop on transmission

University of Malaya

between other stations. The IEEE 802.11 manages to address these problems. It requires and authentication exchange for each station joining in the network in which knowledge of a shared key is demonstrated. It is effective to preventing any of the station that does not have access to the shared key in joining in the network.

The prevention of eaves dropping is achieved using a simple encryption scheme. It masks the contents of transmitted streams of data by combining them with a sequence of random numbers using a bitwise XOR operation. The sequence starts from a shared key and can be reproduced and used to reveal the original data by any station that has knowledge of the key. This stream is called a cipher stream.

University of Malaya

2.5 Operating System

Operating system (OS) is a platform that performs basic tasks, such as recognizing input from the keyboard, sending output to the display screen, keeping track of files and directories on the disk, and controlling peripheral devices such as disk drives and printers.

Besides that, the OS makes sure that different programs and users running at the same time do not interfere with each other. For security, OS ensures that unauthorized users do not access the system. OS provides a software platform to allow application programs run on it.

The most popular operating systems currently are Windows 98, Linux and Windows 2000 Server. While for the PDAs, WinCE is one of the most popular OS there is.

2.5.1 Windows 98

Windows 98 is based on the popular Microsoft Windows 95 Operating System, and is designed for the consumer market. Windows 95/98 was designed for backward compatibility with older DOS and 16bit programs, as well as providing a platform for the newer (back in 1995) 32 bit programs.

Windows 98 works better by making it simple to access the Internet and by providing better system performance along with easier system diagnostics and maintenance. With Windows 98, users' system plays better as well with support for

University of Malaya

the latest graphics, sound, and multimedia technologies, the ability to easily add and remove peripheral devices with support for Universal Serial Bus (USB), and it also enables users to watch TV on PC. Besides that, Windows 98 is compatible with more software (including games) and hardware.

2.5.2 Linux

Linux (often pronounced LIH-nuhks with a short "i") is a UNIX-like operating system that was designed to provide personal computer users a free or very low-cost operating system comparable to traditional and usually more expensive UNIX systems. Linux has a reputation as a very efficient and fast-performing system. Linux's kernel (the central part of the operating system) was developed by Linus Torvalds at the University of Helsinki in Finland. To complete the operating system, Torvalds and other team members made use of system components developed by members of the Free Software Foundation for the GNU project.

Linux is a remarkably complete operating system, including a graphical user interface, an X Window System, TCP/IP, the Emacs editor, and other components usually found in a comprehensive UNIX system. Although copyrights are held by various creators of Linux's components, Linux is distributed using the Free Software Foundation's copyleft stipulations that mean any modified version that is redistributed must in turn be freely available.

University of Malaya

Unlike Windows and other proprietary systems, Linux is publicly open and extendible by contributors. Because it conforms to the Portable Operating System Interface standard user and programming interfaces, developers can write programs that can be ported to other operating systems. Linux comes in versions for all the major microprocessor platforms including the Intel, PowerPC, Sparc, and Alpha platforms. It's also available on IBM's S/390. Linux is distributed commercially by a number of companies. A magazine, *Linux Journal*, is published as well as a number of books and pocket references.

Linux is sometimes suggested as a possible publicly-developed alternative to the desktop predominance of Microsoft Windows. Although Linux is popular among users already familiar with UNIX, it remains far behind Windows in numbers of users.

2.5.3 Windows 2000 Server

Windows 2000 Server provides services that let you build and deploy servers more quickly. The new Configure Your Server Wizard significantly reduces the time it takes to build a server and reduces the likelihood of error. Additional new Wizards reduce the time it takes to create new Web sites, create virtual directories, manage security settings, and manage security certificates. And, with the SysPrep utility (available in the Windows 2000 Server Resource Kit), you can dramatically reduce the time it takes to build completely configured Windows 2000-based servers as compared to installing and configuring those same servers by hand. Another feature

University of Malaya

available through the Resource Kit, the Windows Script Host, includes scripts for a number of commonly used administrative functions, such as logon scripting.

Conclusion

Windows 2000 Server is easier to deploy, configure, and use. It provides centralized, customizable management services to reduce TCO. These management services works with existing management solutions and heterogeneous distributed networks, thus allowing IT departments to get maximum value from their current infrastructure services. Systems administrators, network administrators, support personnel, and end users will benefit from the comprehensive management services built into Windows 2000 Server.

University of Malaya

2.5.4 Windows CE

Windows CE is based on the Microsoft Windows operating system but is designed for including or embedding in mobile and other space-constrained devices. Although Microsoft does not explain the "CE," it is reported to have originally stood for "Consumer Electronics." Windows CE is used in several brands of handheld computers and as part of cable TV set-top boxes built for TCI. It competes with EPOC and also with similar operating systems from 3Com (for its PalmPilot) and other companies. Like the full-scale Windows systems, Windows CE is a 32-bit multitasking, multithreading operating system. Microsoft emphasizes that the system was "built from scratch" while taking advantage of Windows architectural concepts and interfaces. Microsoft argues that Windows desktop system users will find that products with Windows CE provide a familiar user interface.

In addition to handheld computers and cable TV boxes, Windows CE is also offered as the operating system for the Auto PC, Microsoft's concept of controlling applications (such as selecting radio channels) while driving, using interactive speech technology.

University of Malaya

2.6 *Database Server*

A database is a structured collection of data. To add, access, and process data stored in a computer database, a database server is needed. There are several database servers available currently: Oracle, MySQL and SQL server 2000.

2.6.1 Oracle

Oracle9*i* Database is the state of the art in object-relational databases. Voted Editors Choice by PC magazine and the #1 database for Linux by Linux Journal, Oracle9*i* Database is the most scalable and full featured database available. Whether driving your web site, packaged applications, data warehouses or OLTP applications, Oracle9*i* Database is a foundation technology for any professional computing environment.

Oracle can runs on UNIX, Linux and Windows platform. However, it is expensive and separate licenses are required for each of its database engine.

2.6.2 MySQL

MySQL is a relational database management system. MySQL stores data in separate tables rather than putting all the data in one big storeroom. This adds speed and flexibility. The tables are linked by defined relations making it possible to combine data from several tables on request.

University of Malaya

MySQL is a small, compact, easy to use database server, ideal for small and medium sized applications. It is client/server implementation that consists of a server and many different client programs. It is available on a variety of UNIX platforms, Linux, Windows NT, Windows 95/98 and Windows 2000.

MySQL is Open Source Software. Open Source means that it is possible for anyone to use and modify. Anybody can download MySQL from the Internet and use it without paying anything. Anybody can study the source code and change it to fit their needs.

2.6.3 SQL Server 2000

Business today demands a different kind of database solution. Performance, scalability, and reliability are essential, and time to market is critical. Beyond these core enterprise qualities, SQL Server 2000 provides agility to your data management and analysis, allowing your organization to adapt quickly and gracefully to derive competitive advantage in a fast-changing environment. From a data management and analysis perspective, it is critical to turn raw data into business intelligence and take full advantage of the opportunities presented by the Web. A complete database and data analysis package, SQL Server 2000 opens the door to the rapid development of a new generation of enterprise-class business applications that can give your company a critical competitive advantage. The record-holder of important benchmark awards for scalability and speed, SQL Server 2000 is a fully Web-

University of Malaya

enabled database product, providing core support for Extensible Markup Language (XML) and the ability to query across the Internet and beyond the firewall.

University of Malaya

2.7 Data Access Technology

The system will require data access technology to enable communication and access to its various databases. A few of the Microsoft Data access strategy and technology is reviewed and considered.

2.7.1 Universal Data Access (UDA)

UDA is a high-level specification developed by Microsoft for accessing data objects regardless of their structure. The strategy of Universal Data Access is to assure open, integrated, standards-based access to all types of data that is from SQL to non-SQL to even unstructured data across a wide variety of applications, from traditional client/server to the web. The main components of UDA are ADO, OLE DB and ODBC.

2.7.2 ADO (Active Data Object)

Active Data Object (ADO) is the Microsoft's newest high-level interface for data objects that most applications developers will use.

ADO is designed to eventually replace *Data Access Objects (DAO)* and *Remote Data Objects (RDO)*. Unlike RDO and DAO, which are designed only for accessing relational databases, ADO is more general and can be used to access all sorts of

University of Malaya

different types of data, including web pages, spreadsheets, and other types of documents.

ADO provides consistent access to data for creating a front-end database client or middle-tier business object using an application, tool, language, or even an Internet browser. ADO is the single data interface for developers creating 1 to n-tier client/server and Web-based data-driven applications.

2.7.3 ADO.NET

ADO.NET is the strategic application-level interface for providing data access services in the Microsoft .NET Platform. It can be used to access data sources using new .NET data providers as well as existing OLE DB Data Providers using the OLE DB. NET Data Provider. ADO .NET is designed specifically for message based web applications while still providing preferable functionality for other applications architectures. Supporting loosely coupled access to data ADO .NET optimizes data sharing with reducing the number of active connections to database reducing the possibility of multiple users contending for limited resources on the database server.

2.7.4 OLE DB

OLE DB Providers are the data access engines or services, as well as the business logic components that these applications can use in a highly interoperable, component-based environment.

University of Malaya

OLE DB is a set of interfaces that are designed to provide data access to *all* data, regardless of type, format or location. It effectively "componentized" database and related data processing functionality, breaking it up into interoperable components that can run as middleware on the client or server across a wide variety of applications. The OLE DB architecture provides for components such as direct data access interfaces, query engines, cursor engines, optimizers, business rules and transaction managers.

The concept of OLE DB is to explode the database into its basic parts. OLE DB delivers components, external to the database, that provide this typical database functionality in reusable component architecture. And these components, because they are not directly linked to the database itself, can be shared across multiple applications, systems and data stores to provide a higher level, universal interface.

2.7.5 ODBC (Open Database Connectivity)

ODBC is a standard database access method developed by Microsoft Corporation. The goal of ODBC is to make it possible to access any data from any application, regardless of which database management system (DBMS) is handling the data. ODBC manages this by inserting a middle layer, called a database *driver*, between an application and the DBMS. The purpose of this layer is to translate the application's data queries into commands that the DBMS understands. For this to work, both the application and the DBMS must be *ODBC-compliant* -- that is, the application must

University of Malaya

be capable of issuing ODBC commands and the DBMS must be capable of responding to them. Since version 2.0, the standard supports SAG SQL.

University of Malaya

2.8 Language/Scripts

2.8.1 ASP

ASP is a server-side scripting technology. ASP is indeed HTML page with an .asp extension. ASP allows for HTML and a scripting language such as VBScript, JScript or Perl to be interspersed in a Web page. When a browser requests an ASP page, the Web server generates a page with HTML code and sends it back to the browser.

One of the most important features about ASP is that it allows user to easily access data and put it on a Web page. User can simply display data from an ODBC-compliant database, or use ASP to make decisions about what to display on a Web page. User can then format the results in any way that they please.

Another important ASP feature is the ability to use cookies to store and retrieve information. The Request object has a Cookie collection, and user can use this in data processing.

2.8.2 XML

. XML (Extensible Markup Language) is a flexible way to create common information formats and share both the format and the data on the World Wide Web, intranets, and elsewhere. For example, computer makers might agree on a standard or common way to describe the information about a computer product (processor speed,

University of Malaya

memory size, and so forth) and then describe the product information format with XML. Such a standard way of describing data would enable a user to send an intelligent agent (a program) to each computer maker's Web site, gather data, and then make a valid comparison. XML can be used by any individual or group of individuals or companies that wants to share information in a consistent way.

2.8.3 JSP (JavaServer Pages)

JavaServer Pages™ (JSP) is a web-scripting technology that can mix static HTML content with server-side scripting to produce dynamic output. By default, JSP uses Java as its scripting language; however, the specification allows other languages to be used, just as ASP can use other languages (such as JavaScript and VBScript). While JSP with Java will be more flexible and robust than scripting platforms based on simpler languages like JavaScript and VBScript.

JSP provides a number of server-side tags that allow developers to perform most dynamic content operations. So developers who are only familiar with scripting, or even those who are simply HTML designers, can use JSP tags for generating simple output. Advanced scripters or Java developers can also use the tags, or they can use the full Java language if they want to perform advanced operations in JSP pages.

2.9 Authoring Tools

2.9.1 Microsoft .NET Framework

Microsoft .NET is a set of Microsoft software technologies for connecting your world of information, people, systems, and devices. It enables an unprecedented level of software integration through the use of XML Web services: small, discrete, building-block applications that connect to each other—as well as to other, larger applications—via the Internet.

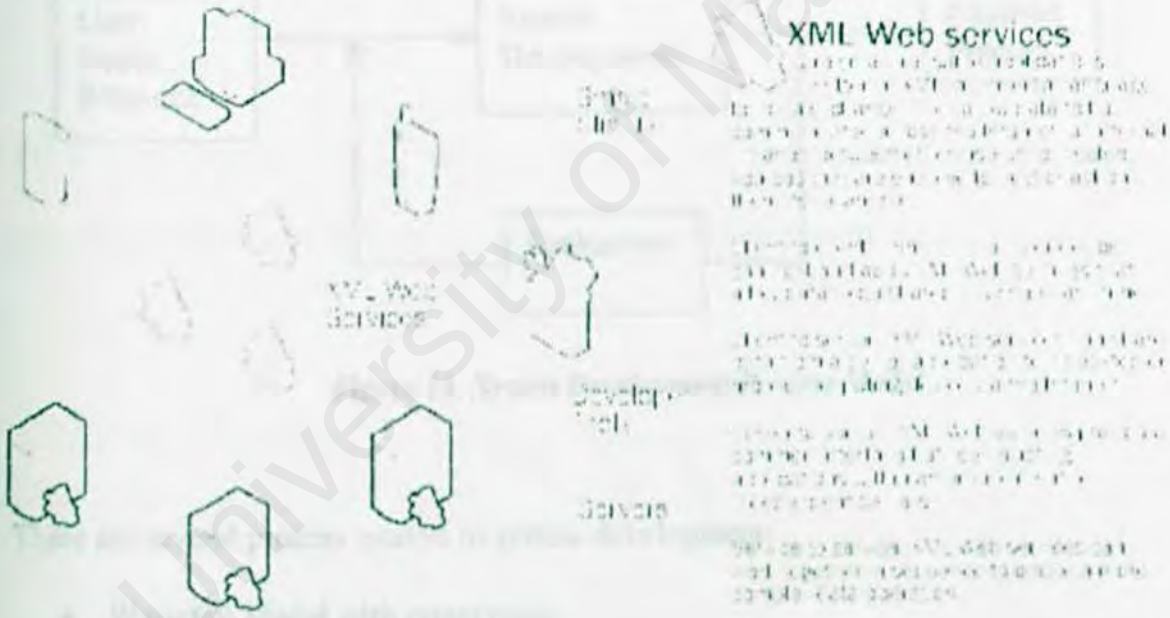


Figure 12: The Components of Microsoft .NET-Connected Software

.NET is infused into the products that make up the Microsoft platform, providing the ability to quickly and reliably build, host, deploy, and utilize secure and connected solutions using XML Web services. The Microsoft platform provides a suite of developer tools, client applications, XML Web services, and servers necessary to participate in this connected world.

University of Malaya

Chapter 3 - System Requirements Analysis

3.1 Methodology

The system development methodology is a method to create a system with a series of steps or operations or can be defined as system life cycle model. Every system development process model (see Figure 13) includes system requirements (user, needs, resource) as input and a finished product as output.

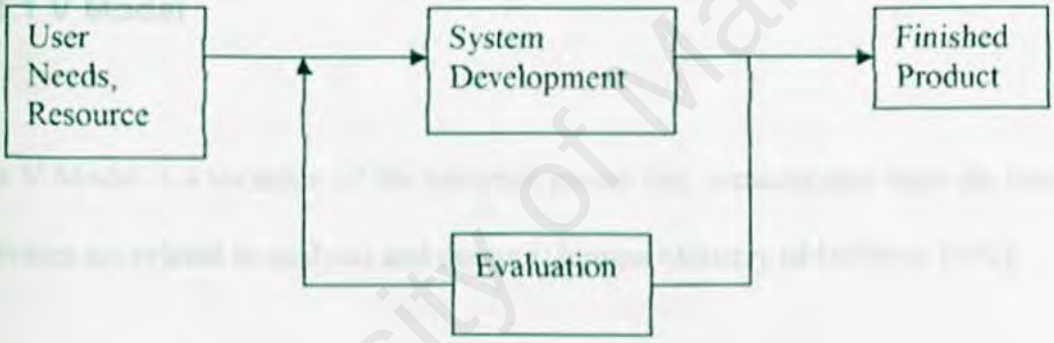


Figure 13: System Development Process Model

There are several process models in system development:

- Waterfall Model with prototyping
- V Model
- System Development Life Cycle (SDLC)
- Spiral Model
- Prototyping Model
- Operational Specification Model
- Transformational Modal

The V Model was chosen for the integrated information system via wifi thesis because:

- It is one of the standard models used in this phase because it is simple to comprehend
- It is a systematic model
- The ability to verify the testing during the development stages is a much welcomed method

3.1.1 V Model

The V Model is a variation of the waterfall model that demonstrates how the testing activities are related to analysis and design (German Ministry of Defence 1992).

University of Malaya

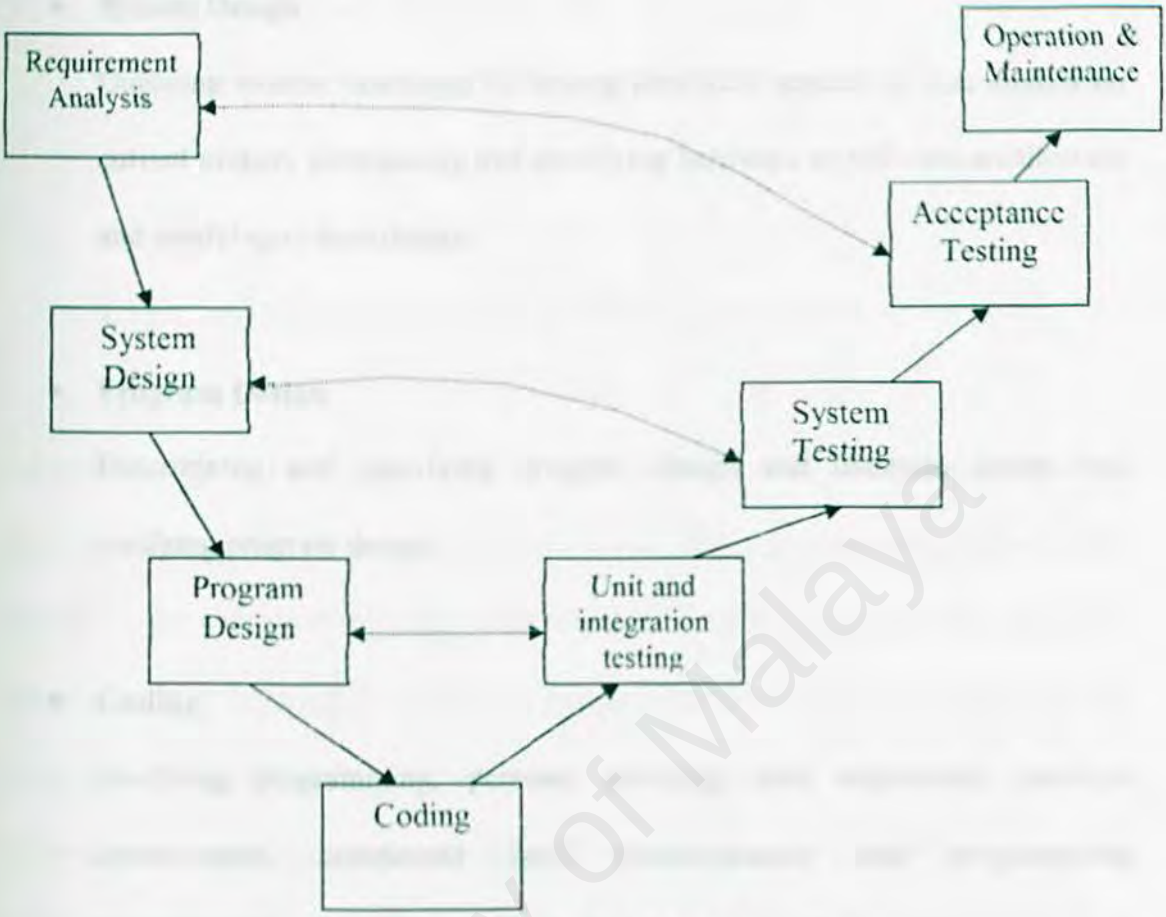


Figure 14: The V-Model

The V-Model consists of 8 stages that are depicted as cascading from on another but in the form of a 'V' shape. The development stages are:

- **Requirements Analysis**

  Understanding and determining users need by having brainstorming, eliciting and analyzing user requirements by having interview, survey or questionnaire session, collecting and specifying all the user requirements and validating requirements.

- **System Design**

  Outlining system functional by having feasibility studies or case studies on current system, determining and specifying hardware or software architecture and verifying system design.

- **Program Design**

  Determining and specifying program design and database design and verifying program design.

- **Coding**

  Involving programming, personal planning, tool acquisition, database development, component level documentation and programming management.

- **Unit and Integration Testing**

  Test units separately and integrate the tested units. Then, testing on the integrated units.

- **System Testing**

  Combining all the integrated units into a system. Testing on the system. Specifying, reviewing and updating of the system test and validating of system.

University of Malaya

- **Acceptance Testing**

  Testing on system completed. The system is delivered.

- **Operation and Maintenance**

  Control and maintain the system. Revalidating of system

As shown in figure 14, coding forms the point of the V, with analysis and design on the left, testing and maintenance on the right. The unit and integration testing addresses the correctness of the program. The V model suggests that unit and integration also be used to verify the program design. That is, during unit and integration testing the coders and test team members should ensure that all aspects of the program design have been implemented correctly in the code. Similarly system testing should verify the system design making sure that all system design is correctly implemented. Acceptance testing validates the requirements by associating a testing step with each elements of the specification; this type if testing checks to see that all the requirements have been fully implemented.

The models linkage of the left side with the right side if the V implies that if problems are found during the verification and validation then the left side of the V can be redone to fix and improve the requirements, design and code before the testing steps on the right side are reenacted. In other words the V model makes more explicit some of the iteration and rework that are hidden in the waterfall depiction. The focus on V is on activity and correctness.

3.1.2 Techniques Used To Define Requirements

In defining requirements for the system there has to be the use of various techniques to gather the nessaccary information regarding it. The research methods that I used are as follows:

3.1.2.1 Interviews

Since this topic has a rather wide area of study there were many people who I interviewed to gather the relevant information regarding certain topics. Among the people who I interviewed were lecturers, technicians and those who are tech savvy in the wireless scene such as programmers and also other students from various private and public institutions.

3.1.2.2 Library and Book Store Research

On doing this thesis I became a frequent visitor at the University Malaya library as well as public bookstores such as the many MPH outlets. I managed to gather a lot of information regarding data integration as well as how the whole wireless scene works and the benefits it offers to people.

3.1.2.3 Internet Research

Perhaps the most useful mode of research was the one obtained in the World Wide Web. There was a lot of information regarding wireless applications and data integration on the net. There were also sufficient tutorials and manuals on the various development tools used for this thesis.

3.1.2.4 Summary of User Requirements

From the research above I found that in order to produce a good integration system via wifi technology it was important to comprehend the necessary protocols involved in the wireless environments as well as the feasibility of it in a certain area. It was important to not overlook the general problems associated with the wireless environment such as hidden stations and fading of sent data or messages. The system expected should be able to:

- Send or receive various data types smoothly over a Wireless LAN
- Able to synchronize data in a Wireless LAN
- Able to access the various kinds of databases available
- Able to provide a platform whereby an offline registration for the faculty would be possible to carry out
- Operate securely with nessasary security measures to prevent security breeches

University of Malaya

3.2 Functional Requirements

Functional requirement is a statement of the service or functions that a system should provide how the system reacts to particular inputs, and how the system should behave in particular situations. **[Sommerville, 1998]**

The functional requirements for this system consist of four main modules. They are the Graphical User Interface, Application Configuration Module, Synchronization Module and Network Module.

3.2.1 Graphical User Interface Module

The Graphical User Interface must provide users an easy means of viewing, editing, and inputting data. The Graphical User Interface must also provide means of editing the application's configuration, inputting the user's details and authentication information and finally the synchronization settings.

3.2.2 Application Configuration Module

The Application Configuration Module maintains user's id and password, host information (IP address, host name, etc), application configuration such as synchronization preferences. Apart from that this module is also in charge of notifying the synchronization module whether to start the synchronization process or not.

3.2.3 Synchronization Module

The Synchronization Module functions to synchronize the local database to and from the server's master database. It also functions to access the local database for data

University of Malaya

retrieval.

Within this module are 2 sub modules. They are the Sync Initialization Sub Module and Data Sync Sub Module

3.2.3.1 Sync Initialization Sub Module

This module directly interfaces with the GUI module to receive and provide data. It accepts new data input into a 'buffer' before passing over to the data sync module to be stored in the local database. This module also receives notification to sync from the Application Configuration Module and it then initiates the Data Sync Module to begin the synchronization process.

3.2.3.2 Data Sync Sub Module

This module directly interfaces with the database. It must have to ability to add, delete, update, and query the database. It must also be able to compare the XML stream of the updates from the remote client. These updates must be checked for conflicts and be resolved either automatically or through the user. Finally, it must also be able to send updated and new data entries in XML format.

3.2.4 Network Module

The Network Module mainly provides the system the ability to network with the server or clients. It also provides means for sending data using a TCP connection and provides an application level of security via data encryption and MAC address authentication. This module is made up of 3 sub modules, namely User

Authentication and Network Initialization sub module, Data Transmission sub module, and Encryption/Decryption sub module.

3.2.4.1 User Authentication and Network Initialization sub module

This module is in charge of initiating the network connection to the server when it detects an available LAN either wired or wireless. The client would first sends an ARP request using the server's known IP address. Upon receiving the MAC address, it then compares it with the known server MAC address stored locally to authenticate the server's IP. Once it is authenticated, it then creates or listens for a TCP connection. After establishing the connection, it then sends over the user id and password. The server side would also repeat the MAC address authentication upon the clients IP used to establish the connection and compare then with the known value for the user id and password. If all those are matched, then the network connection if initialized and a notification of the availability of the network resource send to the application configuration module.

3.2.4.2 Data Transmission sub module

This module merely involves itself in transmitting data through the create TCP connection. It also adds error checking upon the data with the use of checksums upon the transmitted data. Each data packet is within a fixed length and a part of the packet is the checksum value and also the packet's ID. This checksum value is compared and if the packet is invalid, this module rejects it and re-requests the same packet.

3.2.4.3 Encryption/Decryption sub module

This module is in charge of encrypting the XML data before sending it over to the remote side and decrypting it after receiving a XML stream. It uses the DES encryption method using a predefined key (desKEY) and initialization vector (desIV)

3.3 Non-Functional Requirements Analysis

Non-functional specifications are the constraints under which a system must operate and the standards which must be met by the delivered system **[Sommerwille, 1995]**. The Integrated Information System via Wifi Technology must ensures certain application qualities like user-friendliness, correctness, functionality, reliability, flexibility, efficiency as well as maintainability.

3.3.1 User-Friendliness

User interfaces design creates an effective communication medium between a human and a computer. Therefore, it is very important to make sure that the interfaces fulfill user-friendliness so that it would not cause trouble to users. The Golden Rules **[Mandel, 1997]** coins three rules:

3.3.2 Place the user in control

This will define interaction modes in a way that does not force a user into unnecessary or undesired actions. Besides, it also provides flexible interaction for different users for instance via mouse movement and keyboard commands.

3.3.3 Reduce the user's memory load

One of the principles that enable an interface to reduce the user's memory load is by reducing demand on short-term memory. The interface should be designed to reduce the requirements to remember past actions and results.

3.3.4 Make the interface consistent

The interface design should apply to consistent fashion where all visual information must be organized according to a design standard that is maintained throughout all screen displays. Apart from that, input mechanisms are constrained to a limited set that are used consistently throughout the application. Lastly, mechanisms for navigating from task to task are consistently defined and implemented.

3.3.5 Correctness

A program or system must operate correctly or it provides little value to its users. Correctness is the degree to which the software performs its required function. To ensure this application quality, lots of testing and trial-and-errors will be carried out.

3.3.6 Functionality

The functionalities stressed here are the synchronization and transmission of data and it is very important in any application that deals with data retrieval from existing database. Besides, navigation and browsing features as well as application domain-related features will be taken into account.

University of Malaya

3.3.7 Reliability

Reliability is the extent to which a program can be expected to perform its intended function with required precision **[Pressman, 2001]**. It is closely related to correct link processing, error recovery and user input validation and recovery. This quality is essential as it indicates how far users will be confident in the implementation of the new computerized system in getting daily minutes processing done.

3.3.8 Flexibility

For the integrated Information System via Wifi Technology, flexibility of the system is stress on the .NET system which would able users to use different kinds of data to synchronize. It also must take into account the possibility of adding more users into the network and avoid any problems to do with data access and resource sharing/

3.3.9 Efficiency

Undeniable, efficiency is the main key for implementing the new meetings management system. Efficiency is understood as the ability of a process procedure to be called or accessed unlimitedly to produce similar performance outcomes at an acceptable or credible speed **[Sommerwille, 1995]**. Efficiency is measured base on response time performance, page generation speed and graphics generation speed.

University of Malaya

3.3.10 Maintainability

System maintenance accounts would require more effort if the system is not designed according to good programming practices. Maintainability is the ease with which a program can be corrected if an error is encountered, adapted if its environment changes, or enhanced if the customer desires a change in requirements **[Pressman, 2001]** for a particular task or functionality.

3.3.11 Security

The proposed system has also security measures to minimize the risk of data exposure to unauthorized people.

University of Malaya

3.4 Chosen Platform, OS, Database and Tools

3.4.1 Chosen OS

For the Integrated System via Wifi Technology Windows 2000 is chosen as the OS. Microsoft's Windows 2000 is built to work with a series of microprocessors from the Intel Corporation that share the same or similar sets of instructions.

The main reason for choosing Microsoft's Windows 2000 as the development operating system is because most of the computers in FSKTM are currently installed with Windows 2000. Therefore, the implementation of the new system can be done easily and effectively.

Additionally, Windows 2000 Server allows you to configure your network more easily. It provides support for Plug and Play network adapters, significantly reducing device configuration time. It provides services that manage the trust relationships between domains in your organization, and it provides automated replication and local caching of DNS and DHCP information so your network is robust and responsive. Moreover, Connection Sharing Wizards provide an out-of-the-box network address management solution for small businesses.

Day-to-day maintenance is easier as well. System administrators can learn to manage systems from a Windows console much more quickly than they can learn to manage Windows NT® Server 4.0 operating systems thanks to the easier delegation, a more

consistent user interface, and simple-to-use integrated tools provided in Windows 2000 Server.

Centralized Management Services

To be efficient, organizations must manage their hardware, software, and applications in the most automated, secure, and "hands off" method possible. To this end, as information technology systems have become more distributed, organizations have invested a great deal of time and resources in building information and management systems to help manage networks, servers, and client platforms. One of the goals of these systems is to deliver a highly reliable level of service. Another goal is to minimize the overall costs of managing the distributed environment. Further, it's imperative, both to users and to administrators, that these systems be well integrated and easy to use.

One of the major costs highlighted in recent reports on total cost of ownership (TCO), is lost desktop productivity caused by user error, such as changing the system configuration and rendering the computer unworkable. Having too many features or nonessential applications installed on the desktop can also distract users. To solve these problems, you need a means to control a user's access to key configuration files as well as to features and applications that are not required to do that user's particular job. To be successful, this means of control must be flexible and customizable—you must be able to control the computer configurations of individuals and groups of users based on user job responsibilities and computer skills.

Windows 2000 Server is designed to address all of these concerns. It provides powerful management services through infrastructure enhancements such as the Active Directory™ service, as well as tools built on the infrastructure, such as IntelliMirror™ management technologies. Windows 2000 Server delivers powerful, comprehensive management services to better manage servers, networks and Windows-based desktops, including:

Windows Script Host (WSH) allows you to automate and integrate common tasks using a variety of scripting environments including Microsoft® Visual Basic®, Scripting Edition (VBScript), Microsoft Jscript®, and Perl. This feature includes direct scripting to Active Directory and WMI.

I have also chosen Windows CE as the other platform simply because it is the most common platform used in PDA devices. The benefits that Windows CE shared is the same as a normal Windows user on a desktop would have and therefore it only made sense to use this platform.

3.4.2 Chosen Database Management System

The main DBMS that have been chosen for the server-side of the system is MS SQL Server 2000.

SQL Server 2000 provides agility to your data management and analysis, allowing your organization to adapt quickly and gracefully to derive competitive advantage in a fast-changing environment. From a data management and analysis perspective, it is critical to turn raw data into business intelligence and take full advantage of the

University of Malaya

opportunities presented by the Web. A complete database and data analysis package, SQL Server 2000 opens the door to the rapid development of a new generation of enterprise-class business applications that can give your company a critical competitive advantage. The record-holder of important benchmark awards for scalability and speed, SQL Server 2000 is a fully Web-enabled database product, providing core support for Extensible Markup Language (XML) and the ability to query across the Internet and beyond the firewall.

Fully Web-Enabled

SQL Server 2000 provides extensive database programming capabilities built on Web standards. Rich XML and Internet standard support give you the ability to store and retrieve data in XML format easily with built-in stored procedures. You can also use XML updategrams to insert, update and delete data easily.

- **Easy access to data through the Web.** With SQL Server 2000, you can use HTTP to send queries to the database, perform full-text search on documents stored in database, and run queries over the Web with natural language.
- **Powerful, flexible Web-based analysis.** SQL Server 2000 Analysis Services capabilities are extended to the Internet. You can access and manipulate cube data by means of a Web browser.

Highly Scalable and Reliable

Achieve unparalleled scalability and reliability with SQL Server 2000. With scale up and scale out capabilities, SQL Server meets the needs of demanding ecommerce and enterprise applications.

University of Malaya

- **Scale up.** SQL Server 2000 takes advantage of symmetrical multiprocessor (SMP) systems. SQL Server Enterprise Edition can use up to 32 processors and 64 GB of RAM.
- **Scale out.** Scale out distributes the database and data load across servers.
- **Availability.** SQL Server 2000 achieves maximum availability through enhanced failover clustering, log shipping, and new backup strategies.

Fastest Time-to-Market

SQL Server 2000 is the data management and analysis backbone of the Microsoft .NET Enterprise Servers. SQL Server 2000 includes tools to speed development from concept to final delivery.

- **Integrated and extensible analysis services.** With SQL Server 2000, you can build end-to-end analysis solutions with integrated tools to create value from data. Additionally, you can automatically drive business processes based on analysis results and flexibly retrieve custom result sets from the most complex calculations.
- **Quick development, debugging, and data transformation.** SQL Server 2000 features the ability to interactively tune and debug queries, quickly move and transform data from any source, and define and use functions as if they were built in to Transact-SQL. You can visually design and code database applications from any Visual Studio tool.
- **Simplified management and tuning.** With SQL Server 2000, it is easy to manage databases centrally alongside all enterprise resources. Stay online while easily moving and copying databases across computers or between instances.

University of Malaya

3.4.3 Chosen Development Platform

The .NET Framework was the chosen development platform for this thesis.

The .NET Framework is a new computing platform that simplifies application development in the highly distributed environment of the Internet. The .NET Framework is designed to fulfill the following objectives:

- To provide a consistent object-oriented programming environment whether object code is stored and executed locally, executed locally but Internet-distributed, or executed remotely.
- To provide a code-execution environment that minimizes software deployment and versioning conflicts.
- To provide a code-execution environment that guarantees safe execution of code, including code created by an unknown or semi-trusted third party.
- To provide a code-execution environment that eliminates the performance problems of scripted or interpreted environments.
- To make the developer experience consistent across widely varying types of applications, such as Windows-based applications and Web-based applications.
- To build all communication on industry standards to ensure that code based on the .NET Framework can integrate with any other code.

The .NET Framework has two main components: the common language runtime and the .NET Framework class library. The common language runtime is the foundation

of the .NET Framework. The class library, the other main component of the .NET Framework, is a comprehensive, object-oriented collection of reusable types that you can use to develop applications ranging from traditional command-line or graphical user interface (GUI) applications to applications based on the latest innovations provided by ASP.NET, such as Web Forms and XML Web services.

The following illustration (see figure 15) shows the relationship of the common language runtime and the class library to your applications and to the overall system. The illustration also shows how managed code operates within a larger architecture.

.NET Framework in context

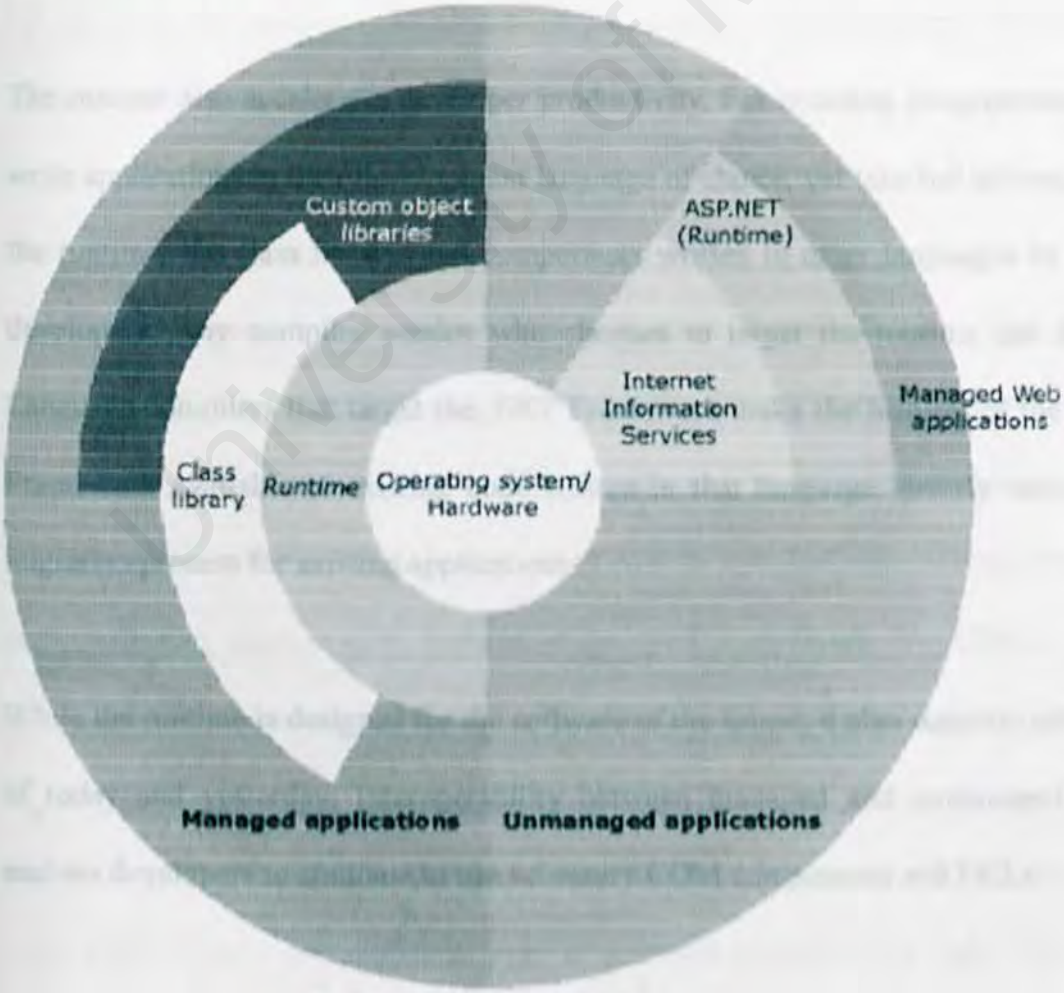


Figure 15: .NET Framework

Features of the Common Language Runtime

The common language runtime manages memory, thread execution, code execution, code safety verification, compilation, and other system services. These features are intrinsic to the managed code that runs on the common language runtime.

With regards to security, managed components are awarded varying degrees of trust, depending on a number of factors that include their origin (such as the Internet, enterprise network, or local computer). This means that a managed component might or might not be able to perform file-access operations, registry-access operations, or other sensitive functions, even if it is being used in the same active application.

The runtime also accelerates developer productivity. For example, programmers can write applications in their development language of choice, yet take full advantage of the runtime, the class library, and components written in other languages by other developers. Any compiler vendor who chooses to target the runtime can do so. Language compilers that target the .NET Framework make the features of the .NET Framework available to existing code written in that language, greatly easing the migration process for existing applications.

While the runtime is designed for the software of the future, it also supports software of today and yesterday. Interoperability between managed and unmanaged code enables developers to continue to use necessary COM components and DLLs.

The runtime is designed to enhance performance. Although the common language runtime provides many standard runtime services, managed code is never interpreted. A feature called just-in-time (JIT) compiling enables all managed code to run in the native machine language of the system on which it is executing. Meanwhile, the memory manager removes the possibilities of fragmented memory and increases memory locality-of-reference to further increase performance.

Finally, the runtime can be hosted by high-performance, server-side applications, such as Microsoft® SQL Server™ and Internet Information Services (IIS). This infrastructure enables you to use managed code to write your business logic, while still enjoying the superior performance of the industry's best enterprise servers that support runtime hosting.

.NET Framework Class Library

The .NET Framework class library is a collection of reusable types that tightly integrate with the common language runtime. The class library is object-oriented, providing types from which your own managed code can derive functionality. This not only makes the .NET Framework types easy to use, but also reduces the time associated with learning new features of the .NET Framework. In addition, third-party components can integrate seamlessly with classes in the .NET Framework.

As you would expect from an object-oriented class library, the .NET Framework types enable you to accomplish a range of common programming tasks, including tasks such as string management, data collection, database connectivity, and file

University of Malaya

access. In addition to these common tasks, the class library includes types that support a variety of specialized development scenarios. For example, you can use the .NET Framework to develop the following types of applications and services:

- Console applications.
- Scripted or hosted applications.
- Windows GUI applications (Windows Forms).
- ASP.NET applications.
- XML Web services.
- Windows services.

Client Application Development

Client applications are the closest to a traditional style of application in Windows-based programming. These are the types of applications that display windows or forms on the desktop, enabling a user to perform a task. Client applications include applications such as word processors and spreadsheets, as well as custom business applications such as data-entry tools, reporting tools, and so on. Client applications usually employ windows, menus, buttons, and other GUI elements, and they likely access local resources such as the file system and peripherals such as printers.

Another kind of client application is the traditional ActiveX control (now replaced by the managed Windows Forms control) deployed over the Internet as a Web page. This application is much like other client applications: it is executed natively, has access to local resources, and includes graphical elements.

University of Malaya

In the past, developers created such applications using C/C++ in conjunction with the Microsoft Foundation Classes (MFC) or with a rapid application development (RAD) environment such as Microsoft® Visual Basic®. The .NET Framework incorporates aspects of these existing products into a single, consistent development environment that drastically simplifies the development of client applications.

The Windows Forms classes contained in the .NET Framework are designed to be used for GUI development. You can easily create command windows, buttons, menus, toolbars, and other screen elements with the flexibility necessary to accommodate shifting business needs.

Server Application Development

Server-side applications in the managed world are implemented through runtime hosts. Unmanaged applications host the common language runtime, which allows your custom managed code to control the behavior of the server. This model provides you with all the features of the common language runtime and class library while gaining the performance and scalability of the host server.

The following illustration shows a basic network schema with managed code running in different server environments. Servers such as IIS and SQL Server can perform standard operations while your application logic executes through the managed code.

University of Malaya

Server-side managed code

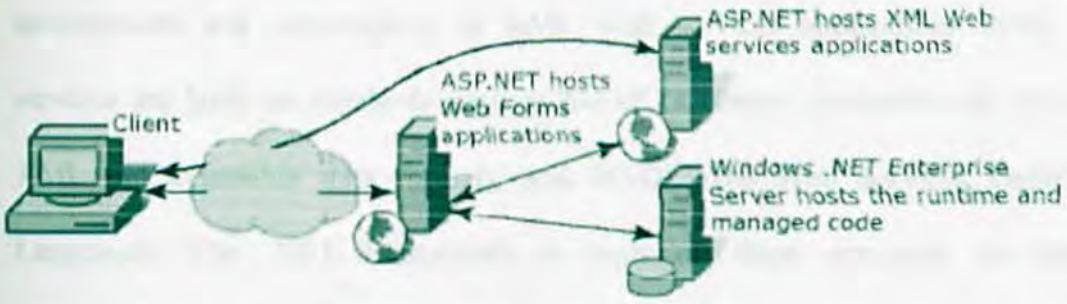


Figure 16: ASP.NET server-side managed code diagram

ASP.NET is the hosting environment that enables developers to use the .NET Framework to target Web-based applications. However, ASP.NET is more than just a runtime host; it is a complete architecture for developing Web sites and Internet-distributed objects using managed code. Both Web Forms and XML Web services use IIS and ASP.NET as the publishing mechanism for applications, and both have a collection of supporting classes in the .NET Framework.

XML Web services, an important evolution in Web-based technology, are distributed, server-side application components similar to common Web sites. However, unlike Web-based applications, XML Web services components have no UI and are not targeted for browsers such as Internet Explorer and Netscape Navigator. Instead, XML Web services consist of reusable software components designed to be consumed by other applications, such as traditional client applications, Web-based applications, or even other XML Web services. As a result, XML Web services technology is rapidly moving application development and deployment into the highly distributed environment of the Internet.

University of Malaya

The .NET Framework also provides a collection of classes and tools to aid in development and consumption of XML Web services applications. XML Web services are built on standards such as SOAP (a remote procedure-call protocol), XML (an extensible data format), and WSDL (the Web Services Description Language). The .NET Framework is built on these standards to promote interoperability with non-Microsoft solutions.

Finally, like Web Forms pages in the managed environment, your XML Web service will run with the speed of native machine language using the scalable communication of IIS.

3.4.4 Chosen Data Access Technology

ADO .NET is the selected data access technology for this thesis. The main reason it is used is because ADO.NET is the strategic application-level interface for providing data access services in the Microsoft .NET Platform. The following points state why ADO .NET was chosen over the largely used ADO.

- Minimized open connections

  When we use ADO .NET it does not depend on continuously live connections to database. This allows us to save resources and is especially important for servers where many users might hog the database.

University of Malaya

- Easy transfer of data through XML

  Data in ADO.NET is transferred through networks in the XML Format and since XML is text-based it is easily transmitted using any protocol and penetrates firewalls

- Datasets

  ADO .NET implements reduced version of the data source.

  It may contain a few tables along with any relationships (unlike recordsets in ADO) and is independent of the data source

  Therefore, the server can work on a few datasets without much performance hit

University of Malaya

Chapter 4 – System Design

4.1 Introduction

System Design is a phase of the waterfall that the entire requirements for the system are translated into system characteristics. The requirements for system are regarding to the analysis that had been discussed in the previous chapter. System design includes the following issues:

- System Architecture Design
- System Functionality Design
- User Interface Design
- Database Design

University of Malaya

4.2 Overview of System Architecture

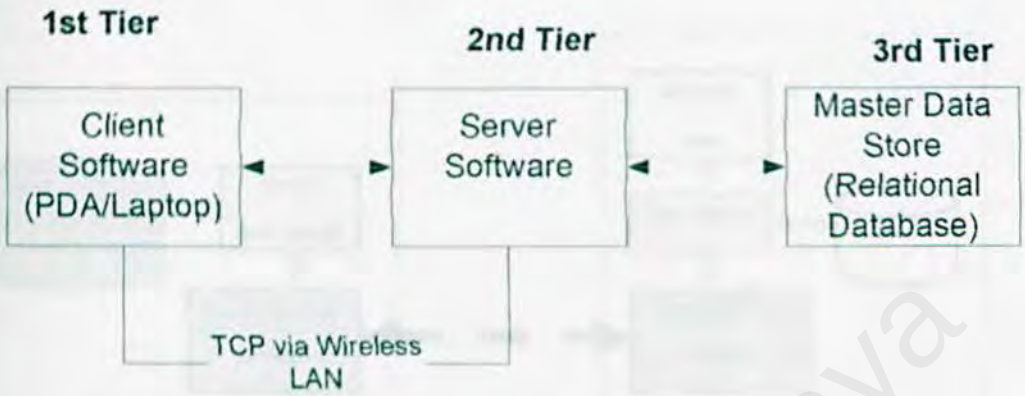


Figure 17: 3-Tier Architecture of Integrated Information System via WiFi Technology

The conceptual architecture of the three-tier application applies when we split an application across three tiers are split into three logical components of the application from a user's point of view: user interface through the client software, synchronization component through the server software and finally the main master data storage. (Refer to Figure 17)

Integrated Information System via WiFi Technology will be using the .NET Framework as the basic development tool of the system development. Essentially, the client side software would be developed using VB.NET for desktop and laptop clients while Embedded Tools for VB 6.0 would be used to develop the client side software for WinCE PDA clients. ADO.NET would be used for all database queries and updates. And finally, data communication would see a TCP connection being use

through a wired LAN or Wireless LAN. A simple illustration on both the ways abovementioned is displayed in Figure 18 and Figure 19.

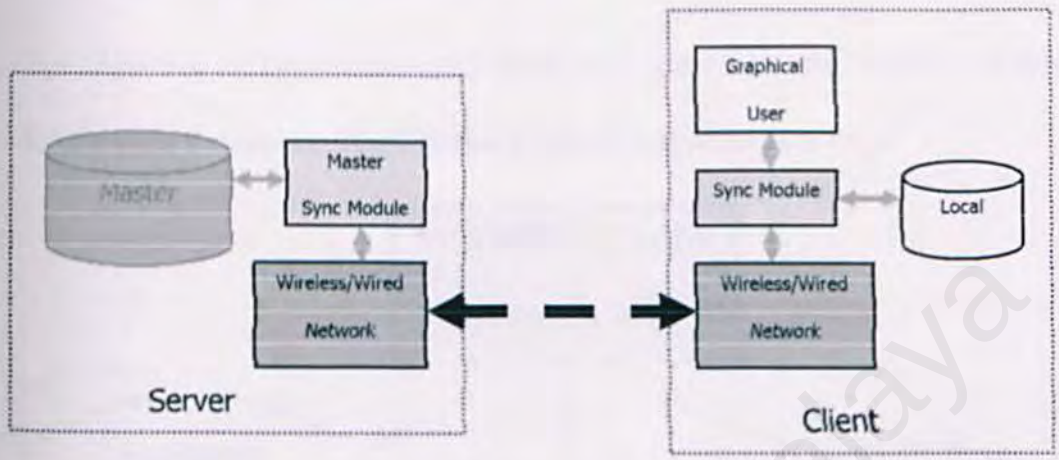


Figure 18: Interaction between the Server and Client Software

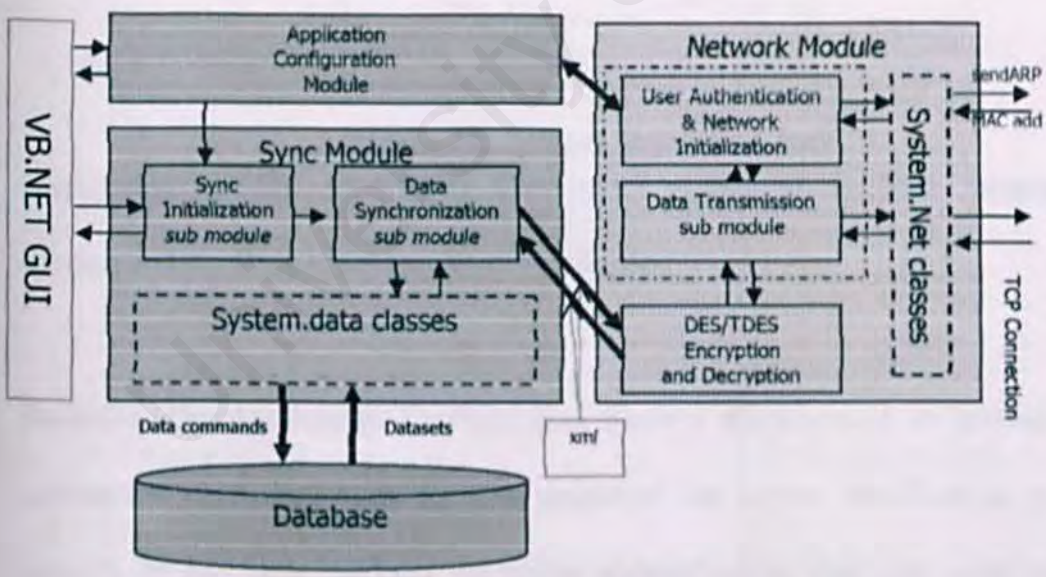


Figure 19: Overview of the system modules.

The main purpose of having three-tier architecture is to assign main functionality to each tier to ensure no function overlapped. Different people could handle each tier using different languages. Therefore, whenever there is error or system fault occurs,

the problems can be detected and easily fixed without interrupting the development of the other tiers.

University of Malaya

4.3 System Functionality Design

4.3.1 System Structure Charts

The objective of system structure chart is to show how the modules in Integrated Information System via WiFi Technology are related to each other.

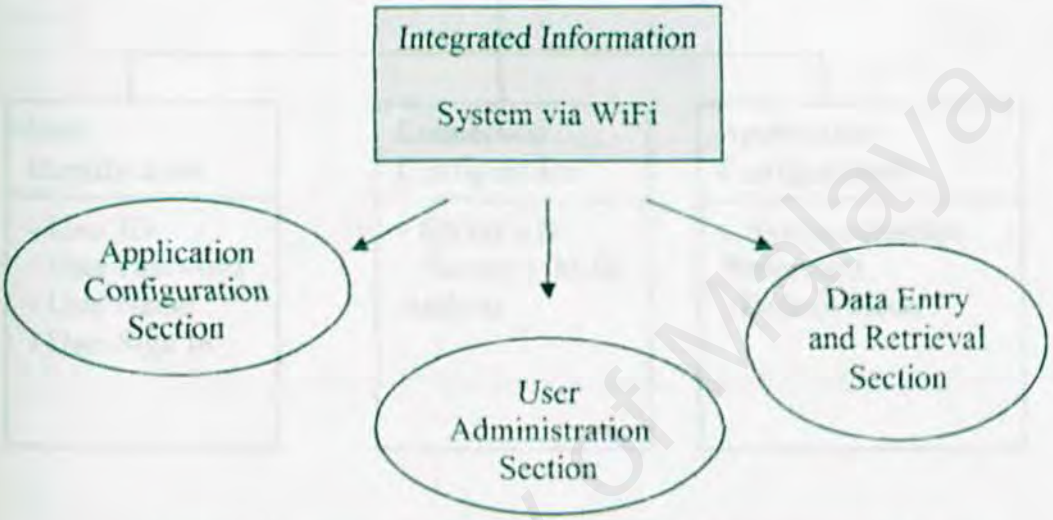


Figure 20: System Structure Chart

The Integrated Information System via WiFi Technology consists of three major parts which are the Application Configuration Section, the User Administration Section and the Data Entry and Retrieval Section.

Basically, the Application Configuration Section allows users to configure their application with their user ID and password for server identification purposes, server's IP and MAC address for server authentication, and their synchronization preferences of the application. The User Administration Section which is available on the server side only is to enable the system administrator to manage and control the use of the systems by their authorized users. Here, the user's MAC address is

registered into the system for authentication and security purposes. Finally, the Data Entry and Retrieval Section allow users to enter their data into the system for later retrieval.

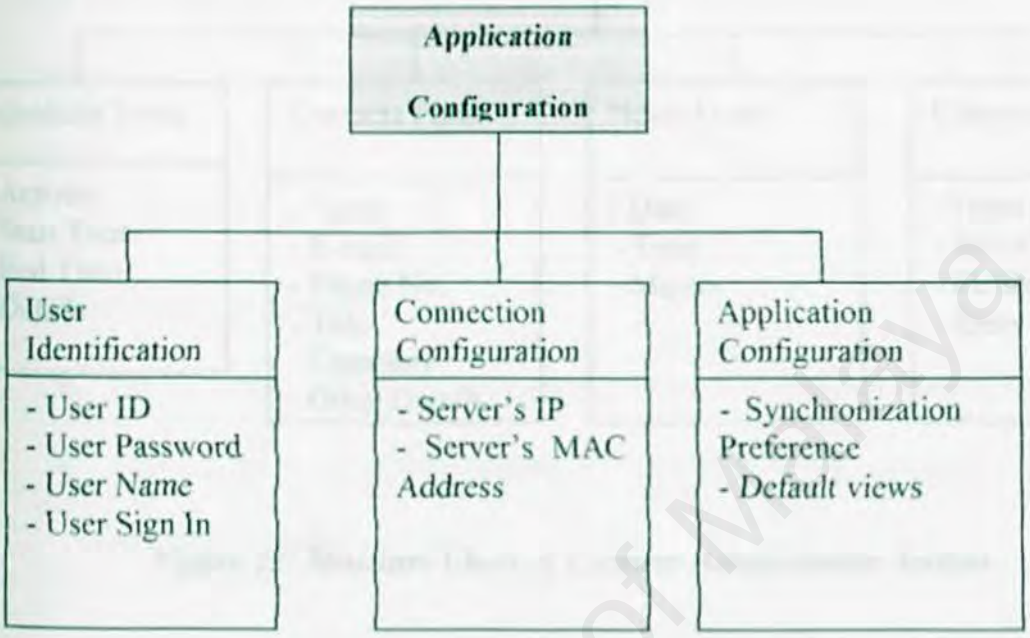


Figure 21: Structure Chart for System Administration Section

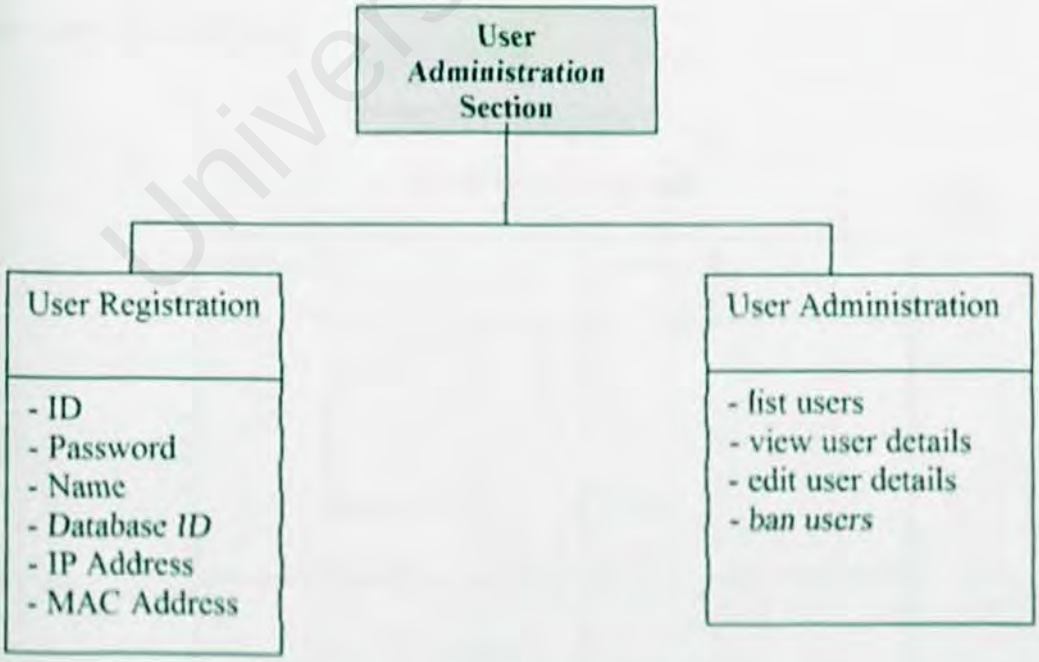


Figure 22: Structure Chart for Student Section

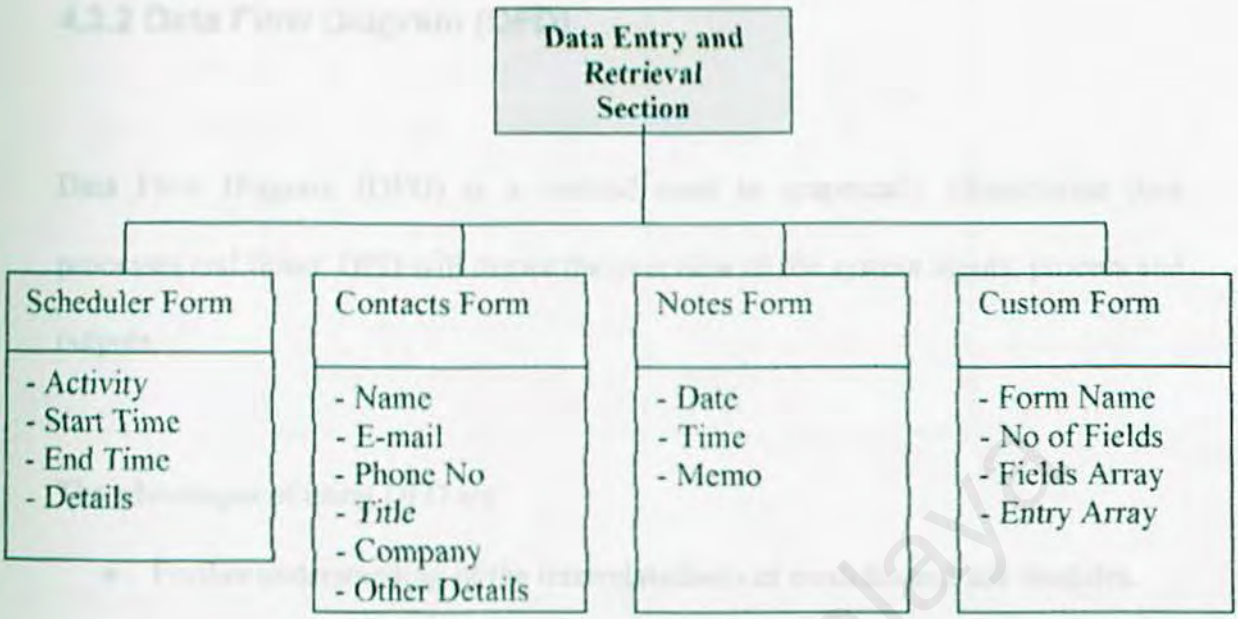


Figure 23: Structure Chart of Lecturer Administrator Section

University of Malaya

4.3.2 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) is a method used to graphically characterize data processes and flows. DFD will depict the overview of the system inputs, process and outputs.

The advantages of using DFD are:

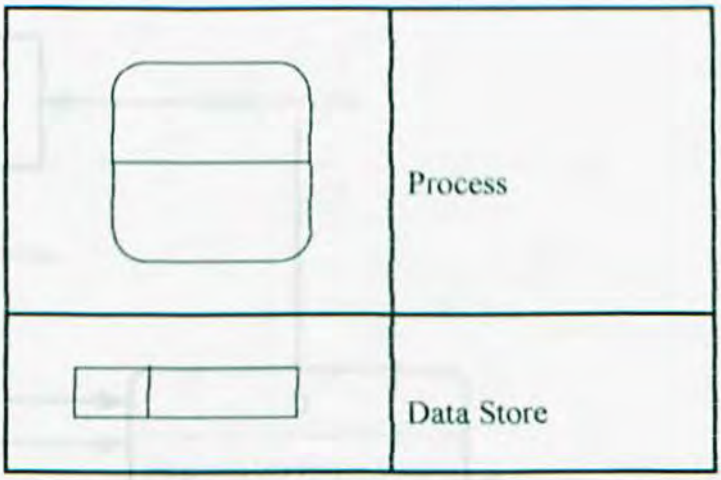
- Further understanding of the interrelatedness of modules and sub modules.
- Analysis of a proposed system to determine if the necessary data and processes have been defined.

DFD is easy to be understood as it has symbols that specify the physical aspects of implementation. There four basic symbols in DFD: entity, flow of data, process and data stores (see Table 3).

Table 3: DFD Symbols

Symbols	Attribute
	Entity
	Flow of Data

University of Malaya



The convention, which is used to design DFD are based on the work by C.Gane and T.Sarson. The data flow is conceptualized with a top-down perspective. So, the Context Level Diagram will be drawn, followed by the Diagram 0. Diagram 0 is an overview process of all the major modules in the *Integrated Information System via WiFi LAN Technology* that includes all the data stores, entities and process involved.

University of Malaya

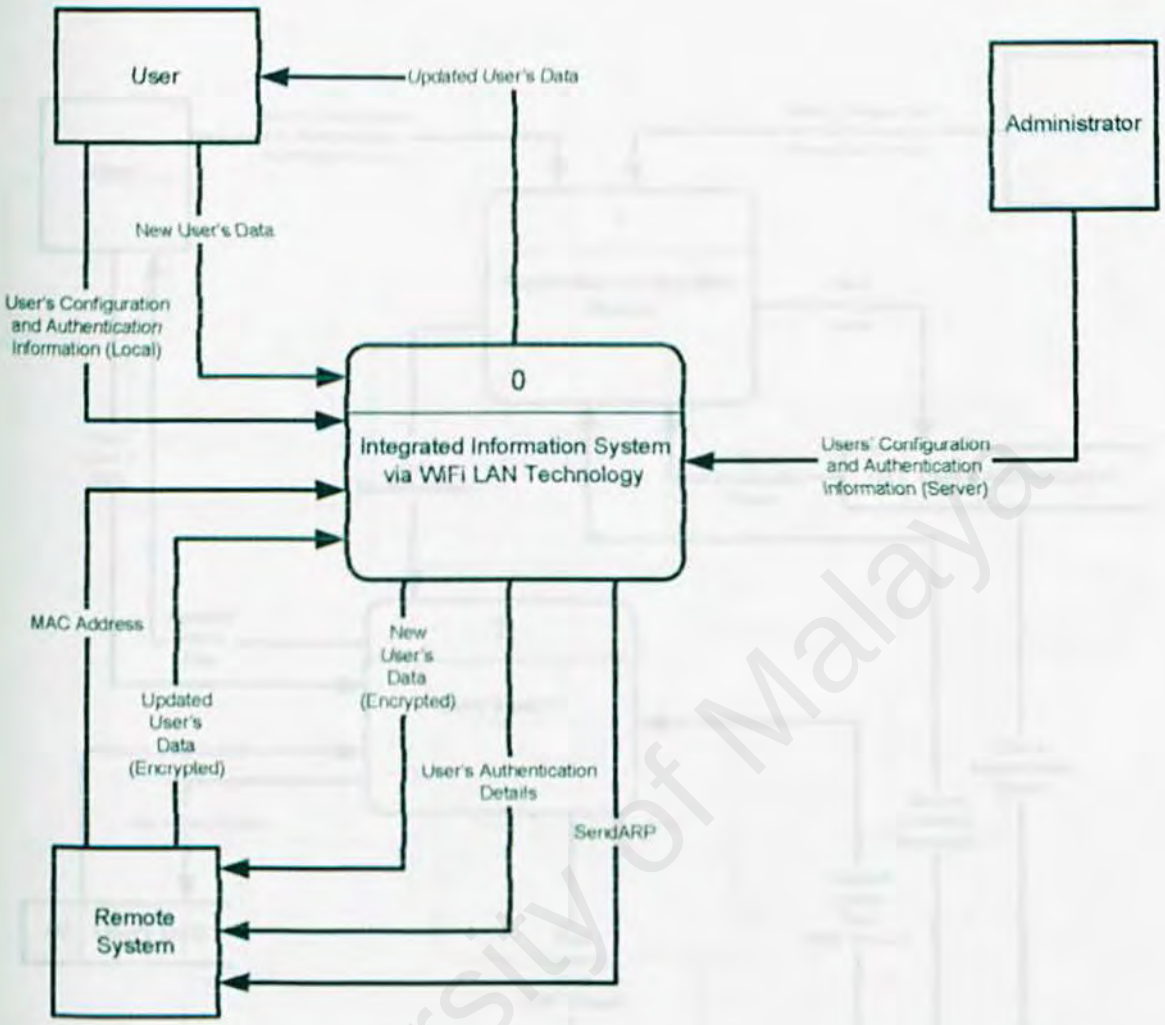


Figure 24: Context Level Diagram

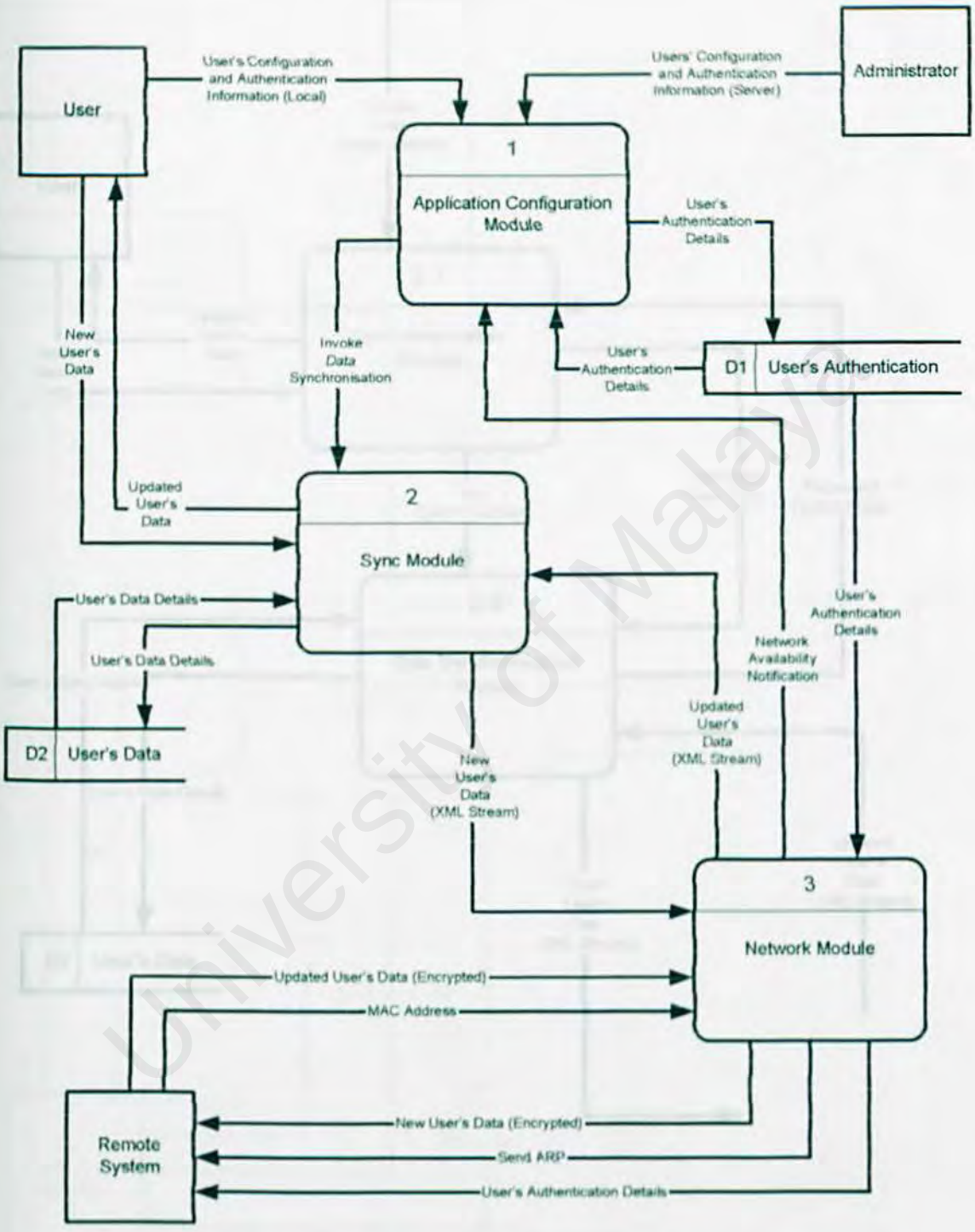


Figure 25: Diagram 0

University of Malaya

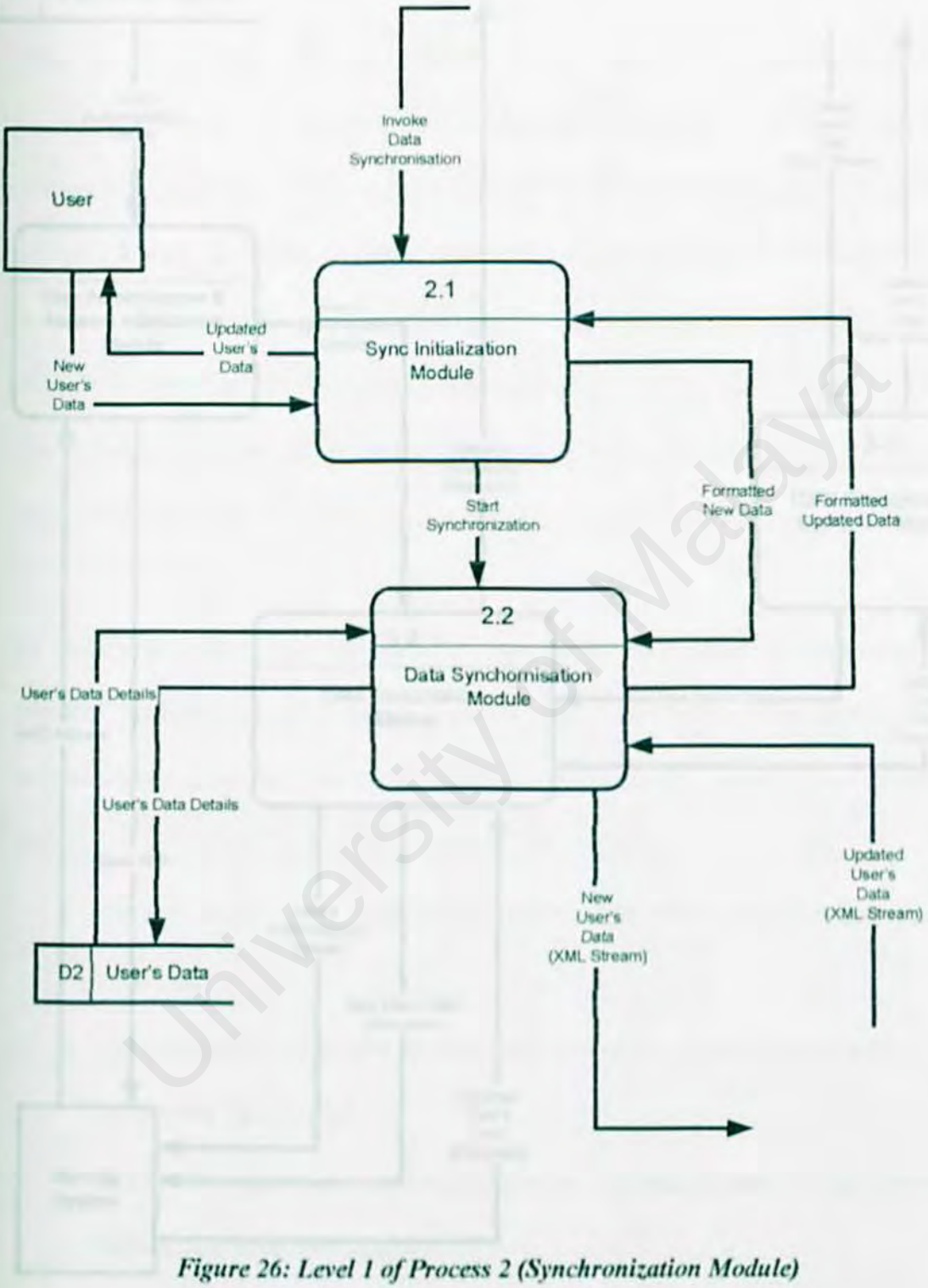


Figure 26: Level 1 of Process 2 (Synchronization Module)

University of Malaya

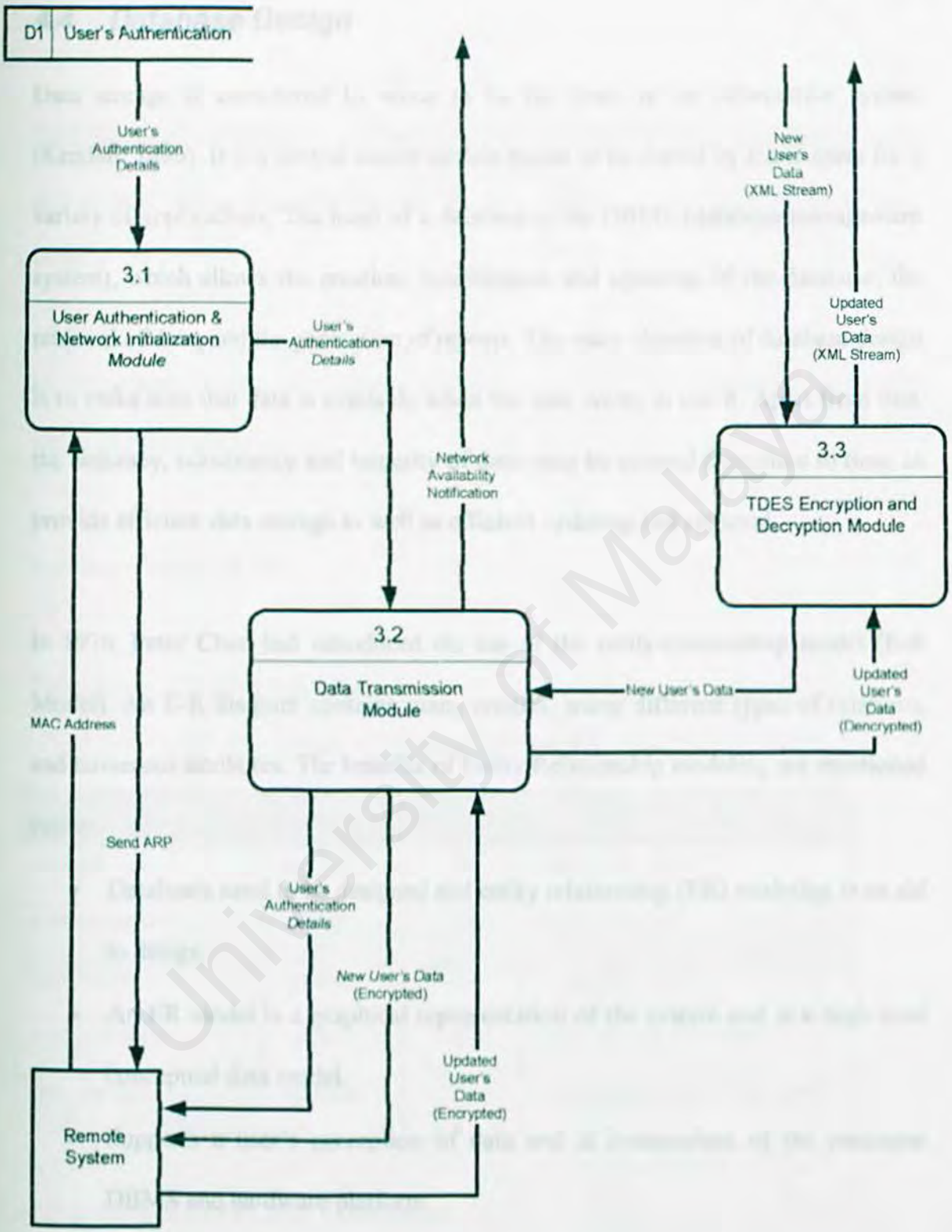


Figure 27: Level 1 of Process 3 (Network Module)

University of Malaya

4.4 Database Design

Data storage is considered by some to be the heart of an information system (Kendall, 1996). It is a central source of data meant to be shared by many users for a variety of applications. The heart of a database is the DBMS (database management system), which allows the creation, modification and updating of the database; the retrieval of data; and the generation of reports. The main objective of database design is to make sure that data is available when the user wants to use it. Apart from that, the accuracy, consistency and integrity of data must be assured from time to time, to provide efficient data storage as well as efficient updating and retrieval.

In 1976, Peter Chen had introduced the use of the entity-relationship model (E-R Model). An E-R diagram contains many entities, many different types of relations, and numerous attributes. The benefits of Entity Relationship modeling are mentioned below:

- Databases need to be designed and entity relationship (ER) modeling is an aid to design.
- An ER model is a graphical representation of the system and is a high-level conceptual data model.
- Supports a user's perception of data and is independent of the particular DBMS and hardware platform.

University of Malaya

4.4.1 Data Dictionary

Data dictionary or metadata can be defined as descriptions of the database structure and contents. Data dictionary defines the field, field type and descriptions of each table.

In this system, a database is created defined as IIS_DB. This database consists of 9 tables and is as the following.

Database Name: **IIS_DB**

Table name: **IISAPPCFG_User**

Table 4: Table of ISSAPPCFG_User

Field Name	Data Type	Length	Note
user_id	varchar	8	User main login ID
user_password	varchar	16	User password
user_name	varchar	30	Name of the user
user_company	varchar	30	Name of user's company
user_regdate	datetime		entry/update time of user datra

Table name: **IISAPPCFG_Connection**

Table 5: Table of ISSAPPCFG_Connection

Field Name	Data Type	Length	Note
server_IP	Varchar	15	Server's IP Address
server _MAC	Varchar	17	Server's IP MAC Address

Table name: **IISAPPCFG_App**

Table 6: Table of IISAPPCFG_App

Field Name	Data Type	Length	Note
view_location	int		Prev last view location 0 = Scheduler 1 = Contacts 2 = Notes > 4 = Other custom forms
sync_type	int		Synchonizing preferences 0 = Always Sync when changed found 1 = Sync only when connected 2 = Only Sync when invoked

Table name: **IISDATA_FormList**

Table 7: Table of IISDATA_FormList

Field Name	Data Type	Length	Note
id	int		Form's ID
name	varchar	10	Name of the form type
fieldno	int		No of fields (max 10)
timestamp	date/time		Time when entry is created
sync	int		Sync status
modified	Data/time		time of update
new	Smallint		To show new entry

University of Malaya

Table name: **IISDATA_Category**

Table 8: Table of IISDATA_Category

Field Name	Data Type	Length	Note
id	int		Category's ID
name	varchar	10	Category Name
timestamp	date/time		Time when entry is created
sync	int		Sync status
modified	Data/time		time of update
new	Smallint		To show new entry

Table name: **IISDATA_FormEntry**

Table 9: Table of IISDATA_FormEntry

Field Name	Data Type	Length	Note
id	int		Entry's ID
form_id	int		Form type
category_id	Int		Category of the entry
field0	memo		Entry for field 0
field1	memo		Entry for field 1
field2	memo		Entry for field 2
field3	memo		Entry for field 3
field4	memo		Entry for field 4
field5	memo		Entry for field 5
field6	memo		Entry for field 6
field7	memo		Entry for field 7
field8	memo		Entry for field 8
field9	memo		Entry for field 9

University of Malaya

timestamp	date/time		Time when entry is created
sync	*int*		*Sync status*
modified	*Data/time*		time of update
new	Smallint		To show new entry

Table name: **IISDATA_ActEntry**

Table 10: Table of IISDATA_FormEntry

Field Name	Data Type	Length	Note
id	int	4	Entry's ID
activity	Varchar	150	Activity
details	memo		Details for the activity
categoryID	Int		Type of the category
start	*Date/time*		*Stating time of the activity*
end	*Date/time*		*Ending time of the activity*
timestamp	date/time		Time when entry is created
sync	int		Sync status
modified	Data/time		time of update
new	Smallint		To show new entry

Table name: **IISDATA_DirEntry**

Table 11: Table of IISDATA_FormEntry

Field Name	Data Type	Length	Note
id	int	4	Entry's ID
Name	Varchar	150	Name of Contact
Position	Varchar	150	Position of Contact

University of Malaya

Company	Varchar	150	Company of Contact
Phone	Varchar	13	Phone No of Contact
Email	Varchar	100	E-mail address of Contact
Address	Varchar	250	Address of Contact
Mobile	Varchar	13	Mobile phone of contact
Others	Memo		Other details
timestamp	date/time		Time when entry is created
sync	int		Sync status
modified	Data/time		time of update
new	Smallint		To show new entry

4.4.2 Relationships - the Class Diagram

There are three types of established inter-table relationships which are one: one (1:1), one: many (1: N) and many: many (M: N). The diagrammatic representation of the system's database relationship is illustrated in the Class diagram in Figure 28.

University of Malaya

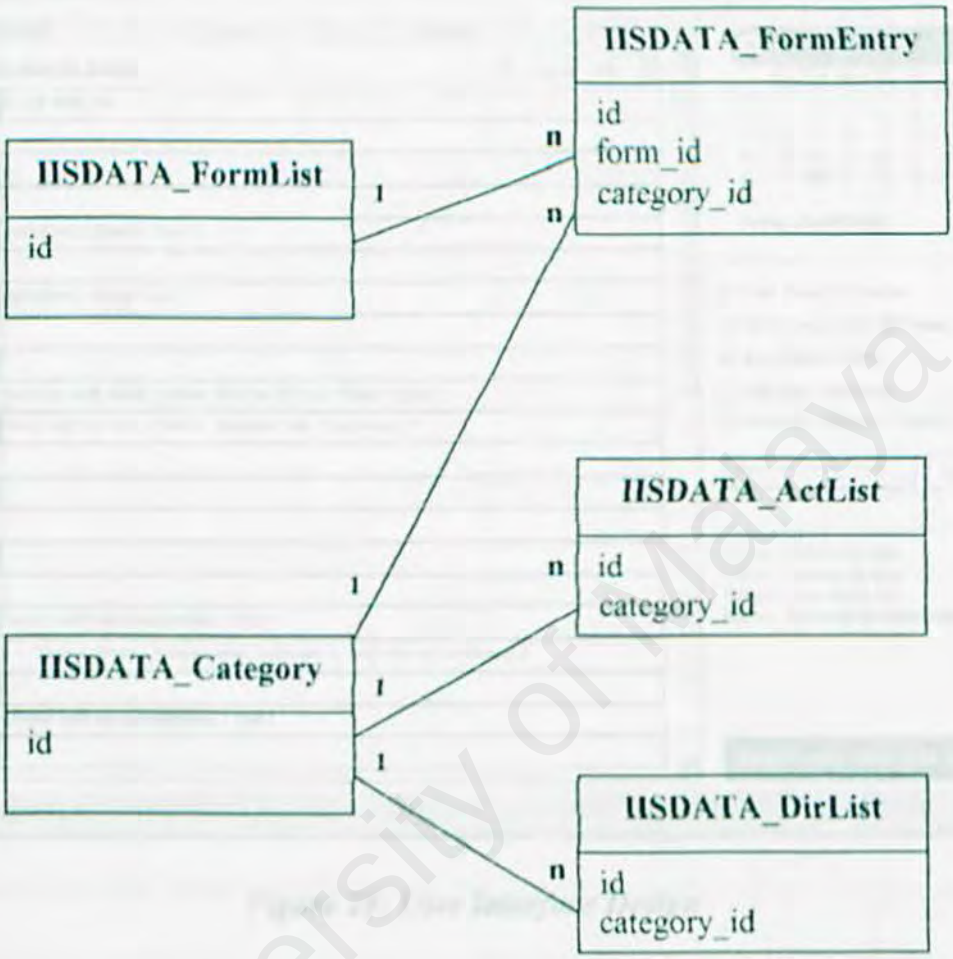


Figure 28: Class diagram of Integrated Information System via WiFi Technology database

University of Malaya

4.5 User Interface Design

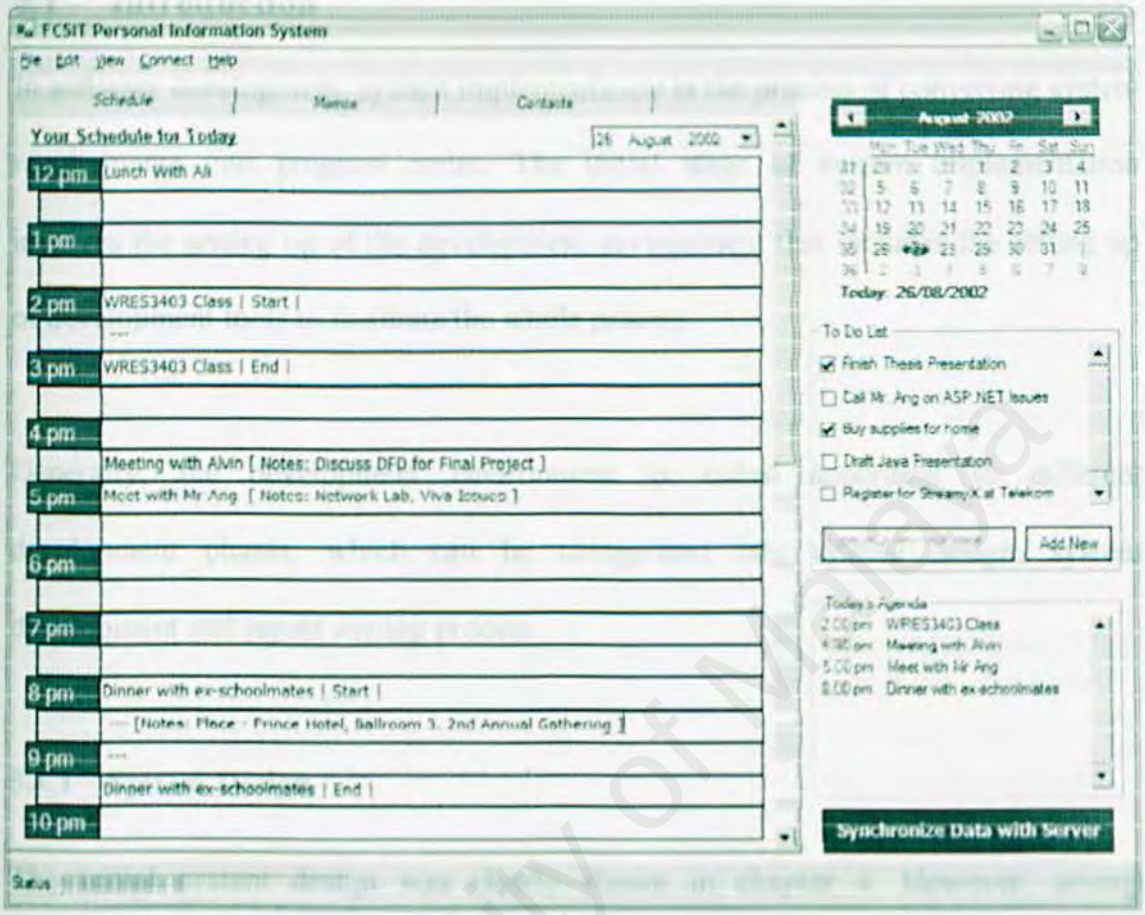


Figure 29: User Interface Design

Chapter 5 - System Implementation

5.1 Introduction

In software development, system implementation is the process of converting system requirements into program codes. The initial stage of system implementation involves the setting up of the development environment that includes the setting up of development tools to facilitate the whole process.

Generally, the development environment is suited according to different development phases, which can be categorized into system design, system development and report writing process.

5.1.1 System Design

The initial system design was clearly shown in chapter 4. However, several considerations and minor modifications needed to be corrected and implemented to the initial system design in order to match the actual needs and requirements.

5.1.2 System Development

The basic tools used for the system development are:

i. Microsoft Window XP Professional (Operating System)

ii. Microsoft Window 2000 Professional (Operating System)

iii. Microsoft SQL Server 2000 (Database Management System)

iv. Microsoft Access XP (Database Management System)

v. Microsoft Visual Studio .NET Enterprise Edition (Integrated Development Environment)

University of Malaya

vi. Microsoft MSDN Library (Reference)

5.1.3 Report Writing

All the problems encountered along with solutions found throughout the processes (from system implementation up till system evaluation) were recorded. Even the results gathered from system testing and system integration were also recorded.

University of Malaya

5.2 System Coding – Coding Approach, Style and Scripting Language

5.2.1 Database Implementation

For IIS, there are basically two main databases. The Server's database is stored locally on the server in which MS SQL Server 2000 is installed. Any data creation, updates or data retrieval is accessed by the IIS Server software via ADO.NET calls with a direct connection to the MS SQL Server 2000 database. As for the Client side, the local database is stored in a MS Access 2000 database format. There is no need for the clients to have MS Access 2000 installed as the IIS Client software directly manipulates the Access database via ADO.NET function calls through the MS JET 4.0 Engine.

As stated within Chapter 4 in the system design, the database on the server side mainly stores the client's information and the authentication information. It also serves to store all the data from its clients for synchronization purposes. The clients' local database on the other hand stores all personal and local data that can be access whether the IIS Client software is connected or not to the server software.

After IIS was completed and tested successfully, all the dummy data were flush from the all databases. All the unnecessary tables were eliminated from ELONS database to avoid data overlapping and to reduce workload of the entire system during deployment.

University of Malaya

5.2.2 Program Implementation

5.2.2.1 Coding Approach

Since the IIS Project development team consists of two individuals, we have chosen to implement both the top-down and bottom-up approach. My partner Alvin Alan Ilangovan took the task to work top-down starting from the GUI of both the Server and Client software where he defined the main functions and sub-functions of the system, while I worked from bottom-up starting from the basic sub modules and then integrating them with Alvin's interface.

Top-down approach is chosen to break the big modules of IIS into functions and procedures. All these small modules or functions are built and developed separately through the bottom-up approach.

Using the bottom-up approach, all the modules and functions are designed using object-oriented methodology. These classes were coded, compiled and tested. Each of them was created to perform only specific functions.

Table 5-1: Example of Object Classes and its Task Performed

Object Class	Task Performed
TryIP.dll	*To retrive IP information from TCP Socket Client*
NetworkConnectDisconnect.dll	To check current network status of the local computer.

5.2.2.2 Coding Style

VB.NET was mainly used to develop the IIS System, both the client and server software. One of the main advantages of working in the .NET platform is the .NET Framework. Within the .NET Framework are many functions and classes that can be used. This effectively cuts down development time as we do not need to code the basic codes from ground-up.

```
Imports System.Net.Sockets
Imports System.Text
Imports System.IO
```

Figure 5-1: .NET Framework Imports example

Interface Programming

Creation of the main GUI interface in VB.NET is relatively simple through the extremely powerful WYSIWYG drag-and-drop form creation interface.

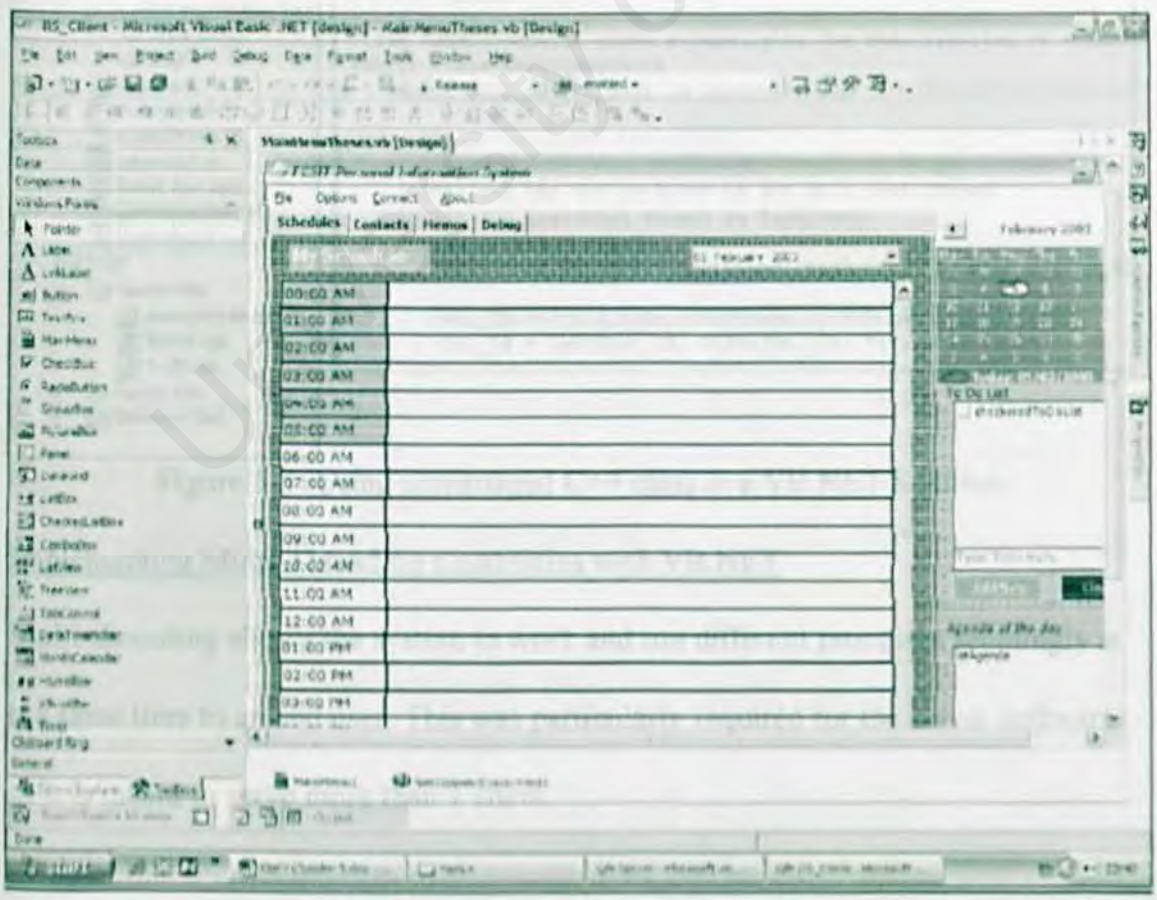


Figure 5-2: Microsoft Visual Studio .NET GUI Tools

Use of Classes

In the .NET development, it's possible to mix and match different languages of source codes to achieve different tasks. During the development of IIS, we had the opportunity to use this technique.

In the IIS Server Software, we some encountered trouble to retrieve the TCP Socket client's IP when using VB.NET. The code that was needed would not compile as managed code. Therefore, the only option was to use an unmanaged C++ Class and get it complied as a DLL which is then referenced from the VB.NET codes!

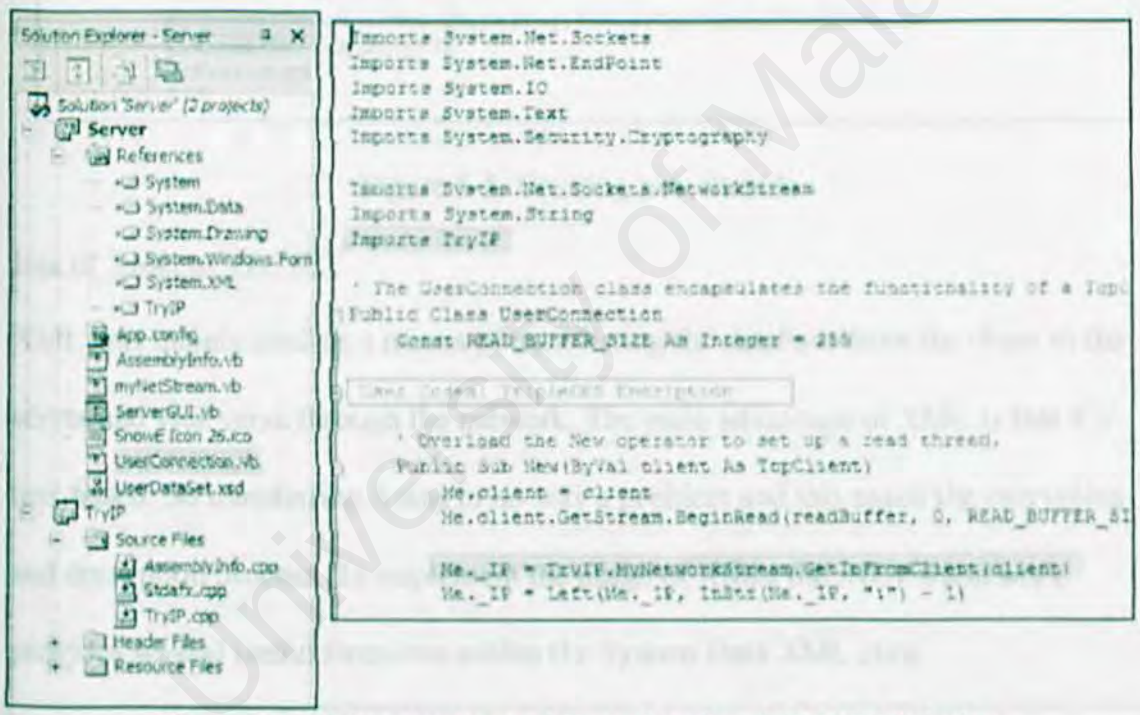


Figure 5-3: Using unmanaged C++ class in a VB.NET Solution

Implementing Multi-Threading capabilities with VB.NET

Multi-threading allows the system to work and run different processes seemingly at the same time to an end user. This was particularly required for the server software as it is meant to serve more than 1 client.

To implement multi-threading in VB.NET, we need to firstly import the System.Threading class.

```
Imports System.Data.SqlClient
Imports System.Configuration.ConfigurationSettings
Imports System.Threading
```

Figure 5-4: Importing System.Threading Class

Then, we just create a new instance of a thread whenever a new TCP connection is created.

```
Private Sub mainForm_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Me.Text = "Network Server"
        ListenerThread = New Threading.Thread(AddressOf DoListen)
        ListenerThread.Start()
        RefreshForm()
    End Sub
```

Figure 5-5: Creating new threads

Use of XML in VB.NET

XML was mainly used as a means of transferring the database from the client to the server and vice versa through the network. The main advantage of XML is that it's text-based. So transferring it was in no way a problem and this eased the encryption and decryption process. To implement the usage of XML, the .NET framework provides several useful functions within the System.Data.XML class

```
    Private Sub ProcessClientData(ByVal sender As UserConnection, ByVal data As
String, ByVal phase As Integer)
        Dim strReader As New StringReader(data)
        Dim XMLReader As New Xml.XmlTextReader(strReader)
        XMLReader.Read()
        Select Case phase
            Case 0
                UpdateStatus("Recieved Data 0 from " & sender.Name)
                remoteDataSet0.ReadXml(XMLReader, XmlReadMode.Auto)
                remoteDataSet0.AcceptChanges()
                ReplyToSender("04", "Sync Successful", sender) 'Sync Success
            Case 1
                UpdateStatus("Recieved Data 1 from " & sender.Name)
                remoteDataSet1.ReadXml(XMLReader, XmlReadMode.Auto)
                remoteDataSet1.AcceptChanges()
                ReplyToSender("14", "Sync Successful", sender) 'Sync Success
```


University of Malaya

```
            Case 2
                UpdateStatus("Recieved Data 2 from " & sender.Name)
                remoteDataSet2.ReadXml(XMLReader, XmlReadMode.Auto)
                remoteDataSet2.AcceptChanges()
                ReplyToSender("24", "Sync Successful", sender) 'Sync Success
            Case 3
                UpdateStatus("Recieved Data 3 from " & sender.Name)
                remoteDataSet3.ReadXml(XMLReader, XmlReadMode.Auto)
                remoteDataSet3.AcceptChanges()
                ReplyToSender("34", "Sync Successful", sender) 'Sync Success
            Case 4
                UpdateStatus("Recieved Data 4 from " & sender.Name)
                remoteDataSet4.ReadXml(XMLReader, XmlReadMode.Auto)
                remoteDataSet4.AcceptChanges()
                BeginSync(sender.DBID)
                ReplyToSender("44", "Sync Successful", sender) 'Sync Success
                RefreshDataSet()
        End Select
    End Sub
```

Figure 5-6: Reading XML data string into Datasets

```
    Private Function getXMLDB(ByVal userDBID As Integer) As String()

        Dim DirCommandString As String = "SELECT * FROM IISDATA_DirEntry WHERE
userid = '" & userDBID & "' ORDER BY dataid ASC"
        Dim ActCommandString As String = "SELECT * FROM IISDATA_ActEntry WHERE
userid = '" & userDBID & "' ORDER BY activityDate ASC"
        Dim FormCommandString As String = "SELECT * FROM IISDATA_FormList WHERE
userid = '" & userDBID & "' ORDER BY dataid ASC"
        Dim FDataCommandString As String = "SELECT * FROM IISDATA_FormEntry WHERE
userid = '" & userDBID & "' ORDER BY dataid ASC"
        Dim CatCommandString As String = "SELECT * FROM IISDATA_Category WHERE
userid = '" & userDBID & "' ORDER BY dataid ASC"

        Dim DirDA As SqlDataAdapter
        DirDA = New SqlDataAdapter(DirCommandString, usrConnection)
        Dim DirDS As New DataSet("Dir")

        Dim ActDA As SqlDataAdapter
        ActDA = New SqlDataAdapter(ActCommandString, usrConnection)
        Dim ActDS As New DataSet("Act")

        Dim FormDA As SqlDataAdapter
        FormDA = New SqlDataAdapter(FormCommandString, usrConnection)
        Dim FormDS As New DataSet("Form")

        Dim FdDA As SqlDataAdapter
        FdDA = New SqlDataAdapter(FDataCommandString, usrConnection)
        Dim FdDS As New DataSet("FdDs")

        Dim CatDA As SqlDataAdapter
        CatDA = New SqlDataAdapter(CatCommandString, usrConnection)
        Dim CatDS As New DataSet("CatDS")

        usrConnection.Open()

        Dim rowcount As Integer
        Try
            DirDA.Fill(DirDS, "Dir")
            ActDA.Fill(ActDS, "Act")
            FormDA.Fill(FormDS, "F")
            FdDA.Fill(FdDS, "Dir")
            CatDA.Fill(CatDS, "Dir")
        Catch exc As Exception
            UpdateStatus("Error: " & exc.message)
            usrConnection.Close()
        End Try

        Dim XMLarray(5) As String
        Try
            XMLarray(0) = DirDS.GetXml
            XMLarray(1) = FormDS.GetXml
```

University Of Malaya

```
            XMLarray(2) = FdDS.GetXml
            XMLarray(3) = CatDS.GetXml
            XMLarray(4) = ActDS.GetXml
        Catch exc As Exception
            UpdateStatus("Error: " & exc.message)
            usrConnection.Close()
        End Try
        usrConnection.Close()
        getXMLDA = XMLarray
    End Function
```

Figure 5-7: Getting XML Data Strings from Dataset

University of Malaya

5.2.2.3 Key Source Codes

Network Module Class

```
Imports System.Net.Sockets
Imports System.Net.EndPoint
Imports System.IO
Imports System.Text
Imports System.Security.Cryptography

Imports System.Net.Sockets.NetworkStream
Imports System.String
Imports TryIP

' The UserConnection class encapsulates the functionality of a TcpClient connection
with streaming for a single user.
Public Class UserConnection
    Const READ_BUFFER_SIZE As Integer = 255


    ' Overload the New operator to set up a read thread.
    Public Sub New(ByVal client As TcpClient)
        Me.client = client
        Me.client.GetStream.BeginRead(readBuffer, 0, READ_BUFFER_SIZE, AddressOf
StreamReceiver, Nothing)

        Me._IP = TryIP.MyNetworkStream.GetIpFromClient(client)
        Me._IP = Left(Me._IP, InStr(Me._IP, ":") - 1)

    End Sub

    Private client As TcpClient
    Private readBuffer(READ_BUFFER_SIZE) As Byte

    ' The Name property uniquely identifies the user connection.
    Private _Name As String
    Public Property Name() As String
        Get
            Return _Name
        End Get
        Set(ByVal Value As String)
            _Name = Value
        End Set
    End Property
    Private _DBID As Integer
    Public Property DBID() As Integer
        Get
            Return _DBID
        End Get
        Set(ByVal Value As Integer)
            _DBID = Value
        End Set
    End Property
    Private _IP As String
    Public Property IP() As String
        Get
            Return _IP
        End Get
        Set(ByVal Value As String)
            _IP = Value
        End Set
    End Property
    Private _pinged As Boolean = False
    Public Property Pinged() As Boolean
        Get
            Return _pinged
        End Get
        Set(ByVal Value As Boolean)
            _pinged = Value
        End Set
    End Property
    Private _PingCount As Integer = 0
    Public Property PingCount() As Integer
        Get
            Return _PingCount
```

University of Malaya

```
        End Get
        Set(ByVal Value As Integer)
            _PingCount = Value
        End Set
    End Property

    Public Event LineReceived(ByVal sender As UserConnection, ByVal Header As String,
ByVal uid As String, ByVal Data As String)
    Public Event UpdateStatus(ByVal Message As String)
    Public Event LostConnection(ByVal userID As String)

    ' This subroutine uses a StreamWriter to send a message to the user.
    Public Sub SendData(ByVal Data As String)
        ' SyncLock ensure that no other threads try to use the stream at the same
time.
        If Data.Length = 2 Then
            Data = Data & "This is a dummy message"
        End If
        Try
            SyncLock client.GetStream
                Dim writer As New IO.StreamWriter(client.GetStream)
                Dim myEncryptData As String = Encrypt(Data)
                RaiseEvent UpdateStatus("SendData -> To Send to client " & vbCrLf &
"Data : " & Data & vbCrLf & "Encrypted: " & myEncryptData)

                writer.Write(myEncryptData & Chr(4))

                ' Make sure all data is sent now.
                writer.Flush()
            End SyncLock
        Catch e As Exception
        End Try
    End Sub

    Dim isFirst As Boolean = True
    Dim prvMessage As String = ""
    Dim strHeader As String
    Dim strUID As String


    ' This is the callback function for TcpClient.GetStream.BeginRead. It begins an
asynchronous read from a stream.
    Private Sub StreamReceiver(ByVal ar As IAsyncResult)
        Dim BytesRead As Integer
        Dim strMessage As String
        Dim encData As String

        Try
            ' Ensure that no other threads try to use the stream at the same time.
            SyncLock client.GetStream
                ' Finish asynchronous read into readBuffer and get number of bytes
read.
                BytesRead = client.GetStream.EndRead(ar)
            End SyncLock
            Dim endChr = Encoding.UTF8.GetString(readBuffer, BytesRead - 1, 1)

            If isFirst And endChr.CompareTo(Chr(4)) = 0 Then
                encData = Encoding.UTF8.GetString(readBuffer, 0, BytesRead - 1)
                encData = Decrypt(encData)

                strHeader = Mid(encData, 1, 2)
                strUID = Mid(encData, 3, 8).TrimEnd
                strMessage = Mid(encData, 11, encData.Length - 10)

                RaiseEvent UpdateStatus("StreamReciever -> Message from Server" &
vbCrLf & "Decrypted: " & strHeader & strUID & strMessage)

                RaiseEvent LineReceived(Me, strHeader, strUID, strMessage)
                isFirst = True
            ElseIf isFirst And Not endChr.CompareTo(Chr(4)) = 0 Then
                encData = Encoding.UTF8.GetString(readBuffer, 0, BytesRead)

                prvMessage = encData
                isFirst = False
            ElseIf Not isFirst And endChr.CompareTo(Chr(4)) = 0 Then
                encData = Encoding.UTF8.GetString(readBuffer, 0, BytesRead - 1)
```

University of Malaya

```
                prvMessage = prvMessage & encData
                prvMessage = Decrypt(prvMessage)
                strHeader = Mid(prvMessage, 1, 2)
                strUID = Mid(prvMessage, 3, 8).TrimEnd
                strMessage = Mid(prvMessage, 11, prvMessage.Length - 10)

                RaiseEvent UpdateStatus("StreamReciever -> Message from Server" &
vbCrLf & "Decrypted: " & strHeader & strUID & strMessage)

                RaiseEvent LineReceived(Me, strHeader, strUID, strMessage)
                isFirst = True
            ElseIf Not isFirst And Not endChr.CompareTo(Chr(4)) = 0 Then
                encData = Encoding.UTF8.GetString(readBuffer, 0, BytesRead)
                prvMessage = prvMessage & encData
                isFirst = False
            End If

            ' Ensure that no other threads try to use the stream at the same time.
            SyncLock client.GetStream
                ' Start a new asynchronous read into readBuffer.
                client.GetStream.BeginRead(readBuffer, 0, READ_BUFFER_SIZE, AddressOf
StreamReceiver, Nothing)
            End SyncLock
        Catch e As Exception
            RaiseEvent UpdateStatus("StreamReciever Error:" & e.Message)
            RaiseEvent LostConnection(Me.Name)
        End Try
    End Sub

    Public Sub New()

    End Sub
End Class
```

Triple DES Encryption and Decryption sub-module

```
#Region " User Codes: TripleDES Encription     "

    Private key() As Byte = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,
17, 18, 19, 20, 21, 22, 23, 24}
    Private iv() As Byte = {65, 110, 68, 26, 69, 178, 200, 219}

    Public Function Encrypt(ByVal plainText As String) As String
        RaiseEvent UpdateStatus("Encrypt Module -> Recieve plainText for encryption")
        RaiseEvent UpdateStatus("Encrypt Module -> Message: " & plainText)
        ' Declare a UTF8Encoding object so we may use the GetByte
        ' method to transform the plainText into a Byte array.
        Dim utf8encoder As UTF8Encoding = New UTF8Encoding()
        Dim inputInBytes() As Byte = utf8encoder.GetBytes(plainText)

        ' Create a new TripleDES service provider
        Dim tdesProvider As TripleDESCryptoServiceProvider = New
TripleDESCryptoServiceProvider()

        ' The ICryptTransform interface uses the TripleDES
        ' crypt provider along with encryption key and init vector
        ' information
        Dim key() As Byte = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,
17, 18, 19, 20, 21, 22, 23, 24}
        Dim iv() As Byte = {65, 110, 68, 26, 69, 178, 200, 219}
        Dim cryptoTransform As ICryptoTransform = tdesProvider.CreateEncryptor(key,
iv)

        ' All cryptographic functions need a stream to output the
        ' encrypted information. Here we declare a memory stream
        ' for this purpose.
        Dim encryptedStream As MemoryStream = New MemoryStream()
        Dim cryptStream As CryptoStream = New CryptoStream(encryptedStream,
cryptoTransform, CryptoStreamMode.Write)
        Dim encryptStreamlength As Integer = 0

        ' Write the encrypted information to the stream. Flush the information
        ' when done to ensure everything is out of the buffer.
        cryptStream.Write(inputInBytes, 0, inputInBytes.Length)
```


University of Malaya

```
        cryptStream.FlushFinalBlock()
        encryptedStream.Position = 0
        encryptStreamlength = CInt(encryptedStream.Length)

        ' Read the stream back into a Byte array and return it to the calling
        ' method.
        Dim result(encryptStreamlength - 1) As Byte
        encryptedStream.Read(result, 0, encryptStreamlength)
        cryptStream.Close()
        Dim strResult As String = Convert.ToBase64String(result)
        Return strResult
        RaiseEvent UpdateStatus("Encrypt Module -> Finished encrypting plainText")
        RaiseEvent UpdateStatus("Encrypt Module -> Encrypted Message: " & strResult)
    End Function
    Public Function Decrypt(ByVal cypherText As String) As String
        RaiseEvent UpdateStatus("Decrypt Module -> Recieve cypherText for
decryption")
        RaiseEvent UpdateStatus("Decrypt Module -> Encrypted Message: " & cypherText)
        ' UTFEncoding is used to transform the decrypted Byte Array
        ' information back into a string.
        Dim utf8encoder As UTF8Encoding = New UTF8Encoding()
        Dim inputInBytes() As Byte = Convert.FromBase64String(cypherText)

        Dim tdesProvider As TripleDESCryptoServiceProvider = New
TripleDESCryptoServiceProvider()

        ' As before we must provide the encryption/decryption key along with
        ' the init vector.
        Dim key() As Byte = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,
17, 18, 19, 20, 21, 22, 23, 24}
        Dim iv() As Byte = {65, 110, 68, 26, 69, 178, 200, 219}
        Dim cryptoTransform As ICryptoTransform = tdesProvider.CreateDecryptor(key,
iv)

        ' Provide a memory stream to decrypt information into
        Dim decryptedStream As MemoryStream = New MemoryStream()
        Dim cryptStream As CryptoStream = New CryptoStream(decryptedStream,
cryptoTransform, CryptoStreamMode.Write)
        Dim decryptStreamlength As Integer = 0

        cryptStream.Write(inputInBytes, 0, inputInBytes.Length)
        cryptStream.FlushFinalBlock()
        decryptedStream.Position = 0
        decryptStreamlength = CInt(decryptedStream.Length)

        ' Read the memory stream and convert it back into a string
        Dim result(decryptStreamlength - 1) As Byte
        decryptedStream.Read(result, 0, decryptStreamlength)
        cryptStream.Close()
        Dim myutf As UTF8Encoding = New UTF8Encoding()
        Dim myResult As String = myutf.GetString(result)

        RaiseEvent UpdateStatus("Decrypt Module -> Finish decrypting cypherText")
        RaiseEvent UpdateStatus("Decrypt Module -> Decrypted Message: " & myResult)

        Return myResult
    End Function

#End Region
```

Retrieving IP Address from TCP Socket Client (TryIP.cpp)

```
// This is the main DLL file.

#include "stdafx.h"

#using <System.dll>
#using <mscorlib.dll>
using System::Net::Sockets::NetworkStream;
using System::Net::Sockets::TcpClient;
using System::String;

#include "TryIP.h"

System::Net::Sockets::Socket * TryIP::MyNetworkStream::get_MySocket()
```


University of Malaya

```
{
        return(this->get_Socket());
}

String * TryIP::MyNetworkStream::get_IPAddress(void)
{
        System::Net::Sockets::Socket *soc = get_Socket();
        System::Net::EndPoint        *Endp = soc->get_RemoteEndPoint();

        return(Endp->ToString());
}

String *TryIP::MyNetworkStream::GetIpFromClient( TcpClient *client )
{
        NetworkStream *stream = client->GetStream();
        MyNetworkStream * myStream = static_cast<MyNetworkStream *>(stream);
        return myStream->get_IPAddress();
}
```

Synchronization sub-module

```
Function BeginSync() As Integer

 Try
  Dim remoteDataSet As New DataSet()

  Dim localrowCount As Integer = 0
  Dim remoterowCount As Integer = 0
  Dim localPos As Integer = 0

  Dim cnt
  Dim localcnt As Integer = 0
  Dim remotecnt As Integer = 0
  Dim fndlocal As Boolean = False

  Dim localDataID As Long
  Dim remoteDataID As Long
  Dim localActDate As String
  Dim remoteActDate As String

  'DirEntry items
  Dim dirname As String
  Dim dirposition As String
  Dim dircompany As String
  Dim dirphone As String
  Dim diremail As String
  Dim diraddress As String
  Dim dirmobile As String
  Dim dirothers As String
  Dim dircategoryid As Long
  Dim dirmodified As DateTime

  'ActEntry items
  Dim activityDate As String
  Dim activity(24) As String
  Dim actmodified As DateTime

  'FormList items
  Dim frmname As String
  Dim frmfieldno As Integer
  Dim frmmodified As DateTime

  'FormEntry items
  Dim formid As Integer
  Dim categoryid As Long
  Dim field(10) As String
  Dim fdatamodified As DateTime

  'Cat items
  Dim catname As String
  Dim catmodified As DateTime

  Dim localSync As Integer
```

University of Malaya

```
Dim localNew As Integer

Try
 resetSync(0)
Catch e As Exception
 UpdateStatus("Error: " & e.Message)
End Try

Try
 getLocalDB()
Catch e As Exception
 UpdateStatus("Error: " & e.Message)
End Try

Dim table As Integer
'Sync Dir, FormList, FormEntry, Category Entries
For table = 0 To 3
 localPos = 0
 Select Case table
  Case 0
   remoteDataSet = remoteDataSet0
  Case 1
   remoteDataSet = remoteDataSet1
  Case 2
   remoteDataSet = remoteDataSet2
  Case 3
   remoteDataSet = remoteDataSet3
 End Select
 If remoteDataSet.Tables.Count <> 0 Then
  If remoteDataSet.Tables(0).Rows.Count <> 0 Then
   localrowCount = localDataSet.Tables.Item(table).Rows.Count
   remoterowCount = remoteDataSet.Tables(0).Rows.Count

   For remotecnt = 0 To remoterowCount - 1
    fndlocal = False
    remoteDataID = remoteDataSet.Tables(0).Rows(remotecnt).Item("dataid")

    'Search for same dataid in local database
    For localcnt = 0 To localrowCount - 1
     localDataID = localDataSet.Tables.Item(table).Rows(localcnt).Item("id")
     If localDataID = remoteDataID Then
      fndlocal = True
      localPos = localcnt
      Exit For
     ElseIf localDataID > remoteDataID Then
      Exit For
     End If
    Next

    If fndlocal = False Then
     'check if its new remote entry
     If remoteDataSet.Tables(0).Rows(remotecnt).Item("new") = 1 Or
remoteDataSet.Tables(0).Rows(remotecnt).Item("new") = 99 Then
      Dim remoteNew As Integer =
remoteDataSet.Tables(0).Rows(remotecnt).Item("new")
      Select Case table
       Case 0 'Add Dir entries
        With remoteDataSet.Tables(0).Rows(remotecnt)
         Try
          dirname = .Item("name")
         Catch
          dirname = ""
         End Try
         Try
          dirposition = .Item("position")
         Catch
          dirposition = ""
         End Try
         Try
          dircompany = .Item("company")
         Catch
          dircompany = ""
         End Try
         Try
          dirphone = .Item("phone")
         Catch
```

University of Malaya

```
            dirphone = ""
           End Try
           Try
            diremail = .Item("email")
           Catch
            diremail = ""
           End Try
           Try
            diraddress = .Item("address")
           Catch
            diraddress = ""
           End Try
           Try
            dirmobile = .Item("mobile")
           Catch
            dirmobile = ""
           End Try
           Try
            dirothers = .Item("others")
           Catch
            dirothers = ""
           End Try
           Try
            dircategoryid = .Item("categoryid")
           Catch
            dircategoryid = 1
           End Try
           Try
            dirmodified = .Item("modified")
           Catch
            dirmodified = Now()
           End Try
          End With
          newDirEntry(remoteDataID, dirname, dirposition, dircompany, dirphone,
diremail, diraddress, dirmobile, dirothers, dircategoryid, 1, remoteNew, dirmodified)
         Case 1 'Add FormList Entries
          With remoteDataSet.Tables(0).Rows(remotecnt)
           Try
            frmname = .Item("name")
           Catch
            frmname = "invalidform"
           End Try
           Try
            frmfieldno = .Item("fieldno")
           Catch
            frmfieldno = 0
           End Try
           Try
            frmmodified = .Item("modified")
           Catch
            frmmodified = Now()
           End Try
          End With
          newFormList(remoteDataID, remoteNew, 1, frmname, frmfieldno, frmmodified)
         Case 2 'Add formEntry entries
          With remoteDataSet.Tables(0).Rows(remotecnt)
           Try
            formid = .Item("form_id")
           Catch
            formid = 1
           End Try
           Try
            categoryid = .Item("category_id")
           Catch
            categoryid = 1
           End Try
           For cnt = 0 To 9
            Try
             field(cnt) = .Item("field" & cnt)
            Catch
             field(cnt) = ""
            End Try
           Next
           Try
            fdatamodified = .Item("modified")
           Catch
```

University of Malaya

```
            fdatamodified = Now()
           End Try
          End With
          newFormEntry(remoteDataID, remoteNew, 1, formid, categoryid, field(0),
field(1), field(2), field(3), field(4), field(5), field(6), field(7), field(8),
field(9), fdatamodified)
         Case 3
          With remoteDataSet.Tables(0).Rows(remotecnt)
           Try
            catname = .Item("name")
           Catch
            catname = "Invalid"
           End Try
           Try
            catmodified = .Item("modified")
           Catch
            catmodified = Now()
           End Try
          End With
          newCat(remoteDataID, remoteNew, 1, catname, catmodified)
         End Select
        End If
       ElseIf fndlocal = True Then
        Dim remoteNew As Integer =
remoteDataSet.Tables(0).Rows(remotecnt).Item("new")
        If DateDiff(DateInterval.Second,
localDataSet.Tables.Item(table).Rows(localPos).Item("modified"),
remoteDataSet.Tables(0).Rows(remotecnt).Item("modified")) > 0 Then
         Select Case table
          Case 0 'Update Dir Entry
           With remoteDataSet.Tables(0).Rows(remotecnt)
            Try
             dirname = .Item("name")
            Catch
             dirname = ""
            End Try
            Try
             dirposition = .Item("position")
            Catch
             dirposition = ""
            End Try
            Try
             dircompany = .Item("company")
            Catch
             dircompany = ""
            End Try
            Try
             dirphone = .Item("phone")
            Catch
             dirphone = ""
            End Try
            Try
             diremail = .Item("email")
            Catch
             diremail = ""
            End Try
            Try
             diraddress = .Item("address")
            Catch
             diraddress = ""
            End Try
            Try
             dirmobile = .Item("mobile")
            Catch
             dirmobile = ""
            End Try
            Try
             dirothers = .Item("others")
            Catch
             dirothers = ""
            End Try
            Try
             dircategoryid = .Item("categoryid")
            Catch
             dircategoryid = 1
            End Try
```

University of Malaya

```
            Try
             dirmodified = .Item("modified")
            Catch
             dirmodified = Now()
            End Try
           End With
           updateDirEntry(remoteDataID, dirname, dirposition, dircompany, dirphone,
diremail, diraddress, dirmobile, dirothers, dircategoryid, 1, remotenew, dirmodified)
          Case 1 'Update FormList Entry
           With remoteDataSet.Tables(0).Rows(remotecnt)
            Try
             frmname = .Item("name")
            Catch
             frmname = "invalidform"
            End Try
            Try
             frmfieldno = .Item("fieldno")
            Catch
             frmfieldno = 0
            End Try
            Try
             frmmodified = .Item("modified")
            Catch
             frmmodified = Now()
            End Try
           End With
           updateFormList(remoteDataID, remotenew, 1, frmname, frmfieldno,
frmmodified)
          Case 2 'update FormEntry Entry
           With remoteDataSet.Tables(0).Rows(remotecnt)
            Try
             formid = .Item("form_id")
            Catch
             formid = 1
            End Try
            Try
             categoryid = .Item("category_id")
            Catch
             categoryid = 1
            End Try
            For cnt = 0 To 9
             Try
              field(cnt) = .Item("field" & cnt)
             Catch
              field(cnt) = ""
             End Try
            Next
            Try
             fdatamodified = .Item("modified")
            Catch
             fdatamodified = Now()
            End Try
           End With
           updateFormEntry(remoteDataID, remotenew, 1, formid, categoryid, field(0),
field(1), field(2), field(3), field(4), field(5), field(6), field(7), field(8),
field(9), fdatamodified)
          Case 3
           With remoteDataSet.Tables(0).Rows(remotecnt)
            Try
             catname = .Item("name")
            Catch
             catname = "Invalid"
            End Try
            Try
             catmodified = .Item("modified")
            Catch
             catmodified = Now()
            End Try
           End With
           updateCat(remoteDataID, remotenew, 1, catname, catmodified)
         End Select
         localDataSet.Tables.Item(table).Rows(localPos).Item("sync") = 1
         localDataSet.AcceptChanges()
        End If
       End If
      Next
```

University of Malaya

```
      localrowCount = localDataSet.Tables.Item(table).Rows.Count
      'parse local db...If local SYNC = 0 (not on remote side) AND New = (0) Then
it's a deleted item.
      For localcnt = 0 To localrowCount - 1
       localDataID = localDataSet.Tables.Item(table).Rows(localcnt).Item("id")
       localSync = localDataSet.Tables.Item(table).Rows(localcnt).Item("sync")
       localNew = localDataSet.Tables.Item(table).Rows(localcnt).Item("new")
       If localSync = 0 And localNew = 0 Then
        Select Case table
         Case 0
          delDirEntry(localDataID)
         Case 1
          delFormList(localDataID)
         Case 2
          delFormEntry(localDataID)
         Case 3
          delCat(localDataID)
        End Select
       End If
      Next
     End If
    End If
   Next

   table = 4
   localPos = 0
   'Sync ActEntry
   If remoteDataSet4.Tables.Count <> 0 Then
    localrowCount = localDataSet.Tables.Item(table).Rows.Count
    remoterowCount = remoteDataSet4.Tables(0).Rows.Count
    If remoterowCount <> 0 Then
     For remotecnt = 0 To remoterowCount - 1
      fndlocal = False
      remoteActDate = remoteDataSet4.Tables(0).Rows(remotecnt).Item("activityDate")
      remoteDataID = remoteDataSet4.Tables(0).Rows(remotecnt).Item("dataid")

      For localcnt = 0 To localrowCount - 1
       localActDate =
localDataSet.Tables.Item(table).Rows(localcnt).Item("activityDate")
       localDataID = localDataSet.Tables.Item(table).Rows(localcnt).Item("id")
       If localActDate = remoteActDate Then
        fndlocal = True
        localPos = localcnt
        Exit For
       End If
      Next

      If fndlocal = False Then
       If remoteDataSet4.Tables(0).Rows(remotecnt).Item("new") = 1 Then
        'Add Act entry
        Dim remoteNew As Integer =
remoteDataSet4.Tables(0).Rows(remotecnt).Item("new")
        With remoteDataSet4.Tables(0).Rows(remotecnt)
         Try
          activityDate = .Item("activityDate")
         Catch
          activityDate = Now().ToString("dd/MM/yyyy")
         End Try
         For cnt = 0 To 23
         Try
          activity(cnt) = .Item("activity" & cnt)
         Catch
          activity(cnt) = ""
         End Try
         Next
         Try
          actmodified = .Item("modified")
         Catch
          actmodified = Now()
         End Try

        End With
        newActEntry(remoteDataID, activityDate, activity, 1, remoteNew, actmodified)
       End If
      ElseIf fndlocal = True Then
```

University of Malaya

```
        localDataSet.Tables.Item(table).Rows(localPos).Item("sync") = 1
        localDataSet.AcceptChanges()
        Dim remoteNew As Integer =
remoteDataSet4.Tables(0).Rows(remotecnt).Item("new")
        Dim difference As Long = DateDiff(DateInterval.Second,
localDataSet.Tables.Item(table).Rows(localPos).Item("modified"),
remoteDataSet4.Tables.Item(0).Rows(remotecnt).Item("modified"))
        If difference > 0 Then
         'Update Act Entries
         With remoteDataSet4.Tables(0).Rows(remotecnt)
          Try
           activityDate = .Item("activityDate")
          Catch
           activityDate = Now().ToString("dd/MM/yyyy")
          End Try
          For cnt = 0 To 23
           Try
            activity(cnt) = .Item("activity" & cnt)
           Catch
            activity(cnt) = ""
           End Try
          Next
          Try
           actmodified = .Item("modified")
          Catch
           actmodified = Now()
          End Try
         End With

         updateActEntry(activityDate, activity, 1, remoteNew, actmodified)
        End If
       End If
      Next

      localrowCount = localDataSet.Tables.Item(table).Rows.Count

      For localcnt = 0 To localrowCount - 1
       localDataID = localDataSet.Tables.Item(table).Rows(localcnt).Item("id")
       localSync = localDataSet.Tables.Item(table).Rows(localcnt).Item("sync")
       localNew = localDataSet.Tables.Item(table).Rows(localcnt).Item("new")
       If localSync = 0 And localNew = 0 Then
        'Del Act Entries
        delActEntry(localDataID)
       End If
      Next
     End If
    End If
   Catch exc As Exception
    UpdateStatus("Error: " & exc.Message)
   End Try
   resetSync(1)
  End Function
```

Retriving MAC Address

We have initially proposed to use MAC Address for authentication purposes. However, the codes that were implemented below, although seemed perfectly correct, just did not want to work. Therefore, the final implementation was only IP authentication and not MAC address.

The codes below conforms with all the documentations that we have checked so far, including MSDN library, and therefore requires more testing and researching before we find out what was wrong.

```
    ' TODO: Find out why Error 87(Incorrect Parameter) occure when SendARP is called
    ' NOTE: Codes seems to be consistent with MSDN documentations.
    Private Const NO_ERROR = 0

    Private Declare Function inet_addr Lib "wsock32.dll" _
      (ByVal s As String) As Long
```


University of Malaya

```
Private Declare Function SendARP Lib "iphlpapi.dll" _
  (ByVal DestIP As Long, _
   ByVal SrcIP As Long, _
ByVal pMacAddr As Long, _
ByVal PhyAddrLen As Long) As Long

Private Declare Sub CopyMemory Lib "kernel32" _
   Alias "RtlMoveMemory" _
  (ByVal dst As Long, _
ByVal src As Long, _
   ByVal bcount As Long)


Private Function GetRemoteMACAddress(ByVal sRemoteIP As String, ByVal
sRemoteMacAddress As String) As Boolean
    Try
        Dim dwRemoteIP As Long
        Dim pMacAddr As Long
        Dim bpMacAddr() As Byte
        Dim PhyAddrLen As Long
        Dim cnt As Long
        Dim tmp As String

        'convert the string IP into
        'an unsigned long value containing
        'a suitable binary representation
        'of the Internet address given
        dwRemoteIP = inet_addr(sRemoteIP)

        If dwRemoteIP <> 0 Then

            'set PhyAddrLen to 6
            PhyAddrLen = 6

            'retrieve the remote MAC address
            Dim ErrorMsg
            ErrorMsg = SendARP(dwRemoteIP, 0&, pMacAddr, PhyAddrLen)
            If ErrorMsg = NO_ERROR Then

                If pMacAddr <> 0 And PhyAddrLen <> 0 Then

                    'returned value is a long pointer
                    'to the mac address, so copy data
                    'to a byte array
                    ReDim bpMacAddr(PhyAddrLen)
                    CopyMemory(bpMacAddr(0), pMacAddr, PhyAddrLen)

                    'loop through array to build string
                    For cnt = 0 To PhyAddrLen - 1

                        If bpMacAddr(cnt) = 0 Then
                            tmp = tmp & "00-"
                        Else
                            tmp = tmp & Hex$(bpMacAddr(cnt)) & "-"
                        End If

                    Next

                    'remove the trailing dash
                    'added above and return True
                    If Len(tmp) > 0 Then
                        sRemoteMacAddress = Mid(tmp, 0, Len(tmp) - 1)
                        GetRemoteMACAddress = True
                    End If

                    Exit Function

                Else
                    GetRemoteMACAddress = False
                End If

            Else
                GetRemoteMACAddress = False
            End If  'SendARP
```

University of Malaya

```
            Else
                GetRemoteMACAddress = False
            End If  'dwRemoteIP

        Catch e As Exception
            UpdateStatus("Error: " & e.Message)
        End Try
    End Function
```

Checking Network Status

```
Imports System.Management

<System.ComponentModel.DefaultEvent("NetworkDisconnect")> _
Public Class NetConnectDisconnect
    Inherits System.ComponentModel.Component
    Private WithEvents WMIDisconnect As ManagementEventWatcher
    Private WithEvents WMIConnect As ManagementEventWatcher

    Public Event NetworkDisconnect(ByVal sender As Object, ByVal e As
NetworkEventArgs)
    Public Event NetworkConnect(ByVal sender As Object, ByVal e As NetworkEventArgs)

    Private Sub SetUpWMI()
        WMIDisconnect = _
            New ManagementEventWatcher("SELECT * FROM MSNdis_StatusMediaDisconnect")
        WMIConnect = _
            New ManagementEventWatcher("SELECT * FROM MSNdis_StatusMediaConnect")

        WMIDisconnect.Scope = _
            New ManagementScope("root\WMI")
        WMIConnect.Scope = _
            New ManagementScope("root\WMI")

        WMIDisconnect.Start()
        WMIConnect.Start()
    End Sub

    Private Sub WMIDisconnect_EventArrived(ByVal sender As Object, _
            ByVal e As EventArrivedEventArgs) _
            Handles WMIDisconnect.EventArrived
        'Network Disconnected
        Dim instanceName As String
        instanceName = CStr(e.NewEvent.Properties("instanceName").value)
        RaiseEvent NetworkDisconnect(Me, New NetworkEventArgs(instanceName))
    End Sub

    Private Sub WMIConnect_EventArrived(ByVal sender As Object, _
            ByVal e As EventArrivedEventArgs) _
            Handles WMIConnect.EventArrived
        'Network Connected
        Dim instanceName As String
        instanceName = CStr(e.NewEvent.Properties("InstanceName").Value)
        RaiseEvent NetworkConnect(Me, New NetworkEventArgs(instanceName))
    End Sub

    Public Sub New()
        SetUpWMI()
    End Sub
End Class

Public Class NetworkEventArgs
    Private m_InstanceName As String
    Friend Sub New(ByVal instanceName As String)
        m_InstanceName = instanceName
    End Sub
    Public ReadOnly Property InstanceName() As String
        Get
            Return m_InstanceName
        End Get
    End Property
End Class
```

University of Malaya

Chapter 6 – TESTING

6.1 Introduction

The main function of testing is to check for bugs and mistakes in a program and to judge whether the program is usable in real life application. During the testing process, several methods can be used to fully test the workings of the system. Some of them include the usage of dummy data to simulate the usage of the system. Another method is to use driver software to test sub modules to see if the codes would work or not.

Bottom-up approach is adopted in system testing for IIS. Each module at the lowest level of the system hierarchy is tested individually. Then, all the tested modules would be related to the next module testing. This approach is repeated until all modules are tested successfully.

6.2 Testing Process

In general, the testing process of IIS can be shown in the following figure. All the details will be further explained in subsequent sub-sections.

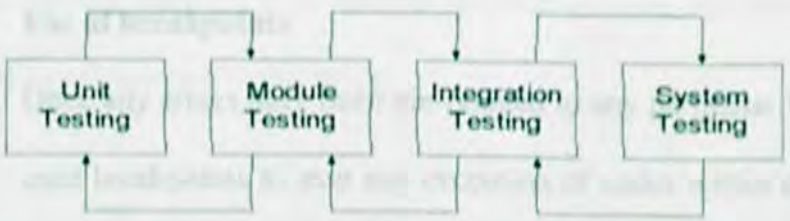


Figure 6-1 Testing Process

University of Malaya

6.2.1 Types of Testing

6.2.1.1 Unit Testing

Unit test is the process to test the individual component to ensure that they function properly. Each component is tested independently without the interference from other system components. Unit test is performed concurrently with the development process.

Techniques used during the process of performing unit testing are as follows:

- **Code Review**

  On of the main features of the MS Visual Studio IDE is the IntelliSense feature that intelligently list down a list of methods or property that is valid. Also, any syntax errors as well as semantic error are highlighted immediately so that correction can be done.

- **Compilation of Codes**

  Compilation of codes would then show all the incorrect calls and invalid techniques used that could not be detected by the real-time code reviewer.

- **Frequently output status messages for debugging purposes**

  If the logical error occurs within any function, then it will be difficult to identify the actual error. Therefore, we make the system to output log messages to

- **Use of breakpoints**

  Once any errors have been pin-pointed to any particular functions, we used breakpoints to stop any execution of codes within the system so that we can trace the codes step by step within the MS Visual Studio

University of Malaya

IDE, which also supplies a 'quick-watch' feature which shows the values of the active and neighboring variables.

6.2.1.2 Module Testing

Module testing is performed without other system modules. A module consists of a collection of dependent components to perform a particular task or function. Different possible test cases are applied to the module and the test results would be verified. Unusual results will be analyzed and they would help in debugging sub-modules in order to produce the desired output.

6.2.1.3 Integration Test

Integration test is needed when all modules are integrated. The main focus in integration test is to navigate the interfaces repeatedly to detect any interface mismatch problem.

Several important aspects are checked to ensure that the flow of the data in IIS is well organized and are user friendly to all the system users.

6.2.1.4 System Test

The sub-systems are integrated to make the entire system. Therefore, the main purpose in system testing is to find errors that result from unanticipated interactions between sub-systems. Besides, it is used to validate whether the system meets its functional and non-functional requirement.

Here, dummy data is created and entered into the databases and multiple testing was done to fully determine all main functions of the system works. Also since IIS is

University of Malaya

based on the client-server model, we had to make sure that all the network communications between server and clients was working perfectly without any flaws. Errors in data transfers between the networks are not tolerable.

Finally, a performance test is performed to compare the integrated modules with the non-functional system requirements. These requirements include security, interoperability, flexibility and reliability.

University of Malaya

Chapter 7 - System Evaluation and Conclusion

7.1 Introduction

Evaluation is the ultimate phase of developing a system and an important phase before delivery the system to the end users. Evaluation was related to user environment, attitudes, information priorities and several other concerns that are to be considered carefully before effectiveness can be concluded. At all phases of the system approaches, evaluation is a process that occurs continuously, drawing on a variety of sources and information.

7.2 Problems Encountered and its Solutions

Debugging the network module

The network module pose to be quite a challenge as there's so many possible outcomes and scenarios for the messages to be listened and sent. Therefore, we had to basically test the network module for several days to test every possible errors and message clashes and then attempt to correct them for each encountered problems.

Initially we did not output debug messages for every action of the network module. This posed to be a major hindrance to debug the network module efficiently, so we finally did just that and debugging was much faster.

Some other problems that we encountered were that sometimes, messages were received faster than the client/server can respond to the previous messages. This problem was just obscure as it only randomly appears and does not always occurs, especially during debug mode when everything is run step-by-step.

University of Malaya

We had to theorize this possibility and we were right when we forced the port listener to 'sleep' for 100 milliseconds after ever read.

Debugging the source codes

While the MS Visual Studio .NET IDE have excellent debugging tools and features, the system sometimes randomly crashes by itself when run in debugging mode, especially when we ran the system step-by-step using after every critical breakpoints we have marked. This proved to be very frustrating.

We then realize that having too many breakpoints and running through too many lines of codes step-by-step causes MS Visual Studio to crash. So, subsequent tests was done with less breakpoints and less code stepping, this however meant that errors could not be directly pin-pointed without going through the code execution several times.

7.3 Evaluation by End User

As IIS is proposed to allow the end-users and their assistants to effectively manage their time, contacts and information, the final stage of system development which is the system testing becomes critical and it needs feedbacks from all respective users in judging the correctness of these functionalities, precise data flow as well as user friendliness of the system's interfaces.

The overall feedback from the beta tester and end users was good and IIS is expected to serve the targeted group well after refining especially the networking modules.

University of Malaya

7.4 System Strengths

- A very user-friendly user interface

  The decision to develop the system as a windows application and not a web-based application proved to be a good choice as we have more control to create a good user interface. All buttons are clearly described and all fields for entry are easily accessible.

- Ability to work offline and online

  Users do not need to be connected to the IIS Server to manage his data and schedules. Each client software has its own local database. This allow them to flexibly use the system any where at any time.

  But once the client gets connected to the server, the data is synchronized to and from the server's database in any case of updates.

- Multiple users sharing the same database

  One of the key strengths of the IIS software is the ability to allow multiple clients to share a single database. This can be set from the IIS Server software and is determined by the administrator of the system.

  The main advantage of this feature is that an assistant may easily update his/her superior's schedules or information directly. And by factoring in the ability to work off and online, this creates an even more flexible working environment for managing data.

- Multi-threaded system

  The IIS server is naturally built to serve more than one client at any one time. Therefore, the port listener needed to be built to support multi-threading.

- Wireless-LAN ready

Implementation of Wireless LAN in the future workplace is inevitable. Therefore, the IIS system was developed with this in mind. The IIS Client software and auto detect whether the client in connected or not to the network. If a network is established, then the software will prompt the user and asks if he wants to connect and synchronize with the server or not.

- Secured data transfer through T-DES data encryption

  The IIS system implements data encryption within the network layer. This is transparent to the end-users which makes the usability of the software much easier while still implementing a strong security to prevent network sniffers and intruders.

- IP Authentication of clients

  We have decided to add extra authentication features other than the usual user id and password. With IP authentication, a user trying to use another PC with an un-authorized IP would not be able to log into the server with the same user id and password.

University of Malaya

7.5 System Constraints and Future Enhancements

The IIS system is still not yet fit to work at its full efficiency. More refining work needs to be done to the system to increase its network reliability. The aspects to be refine and some suggestions to upgrade the system are as below:

1) MAC Address authentication
    a. The needed codes and API was attempted with in the source codes of the project but an error was unresolved although we are sure the SendARP call is consistent with the MSDN and several other documentations.
    b. More testing is required to get the codes working
2) Online Chat
    a. Messages header types can be easily added within the system to process the data as Chat messages and an interface can be added within the system to enable users to chat
3) More forms
    a. Due to the use of a general form fields database with a form type list, more forms can be added without needing changes within the database itself.
    b. Changes is only needed on the interface level
4) More flexible and robust network layer
    a. Current codes works perfectly on the 'best case' scenario. But possibility of failure during the 'worst case' scenario is arises. Such known problems is as following
        i) Client sends or expects a certain messages BUT at exactly the same time the server sends a PING to check for

University of Malaya

user connectivity. This disrupts the client's process flow and the client 'hangs' doing nothing as the server does not reply with the required message.

-OR-

ii) The required messages from clients are sent when the client listener thread timeouts.

b. Due to the limited time and resources, it's not possible to troubleshoot and debug the network layer to achieve 99.9% reliability. Currently, the network failure rate is still relatively high at an estimated of 60-80% working rate

c. More testing time and environment is required to fully debug the network codes to accommodate all possible errors

d. HOWEVER, the user data is no way affected by network failures. Data is not updated / synced if failure in the network is detected.

5) Connectivity to the Internet

a. Theoretically, our network layer should communicate with clients or servers through the Internet if the remote's IP is known

b. However, more testing is required to fully explore this area

c. The Interface could also be enhanced to be able to directly send e-mail using the Contact list's e-mail information

6) Security

a. The current system uses a triple DES shared secret-key for encryption and decryption.

b. Future version could make use of the public/private key for better security.

References

1. B. P. Crow, I. Wadjaja, J. G. Kim, P. T.Sakai, ``IEEE 802.11 Wireless Local Area Networks," *IEEE Commm. Mag.*, vol. 35, no. 9, Sep. 1997, pp. 116-126.
2. E. Prem, Wireless Local Area Networks, Aug 97, http://www.cis.ohio-state.edu/~jain/cis788-97/wireless_lans/index.htm
3. IEEE Standards Wireless Zone, http://standards.ieee.org/wireless/
4. K.C. Chen, "Medium Access Control of Wireless LANs for Mobile Computing," IEEE Network, September/October 1994, pp. 50-63.
5. L. Goldberg, "Wireless LANs: Mobile-Computing's Second Wave," Electronic Design, June 26, 1995
6. R. O. LaMaire, A. Krishna, P. Bhagwat, J. Panian, ``Wireless LANs and Mobile Networking: Standards and Future Directions," *IEEE Commm. Mag.*, vol. 34, no. 8, Aug. 1996, pp. 86-94..
7. T. F. La Porta, K.K. Sabnani, and R.D. Gitlin, "Challenges for Nomadic Computing: Mobility Management and Wireless Communications," Mobile Networks and Applications, Vol. 1, 1996, pp. 3-16.
8. Wireless Local Area Networks, http://www.cis.ohio-state.edu/~jain/cis788-97/wireless_lans/index.htm
9. Wireless Medium Access Control and Physical Layer WG, IEEE Draft Standard P802.11, ``Wireless LAN,"IEEE Stds. Dept., D3, Jan. 1996.
10. Wireless Medium Access Control and Physical Layer WG, IEEE Draft Standard P802.11, ``Wireless LAN,"IEEE Stds. Dept., D3, Jan. 1996.

University of Malaya

Appendix

Research Materials

http://www.starfish.com/solutions/data/data.html

You Are Here: Home > Solutions > Data Synchronization

TrueSync Data Synchronization

TrueSync® is one of the most widely used synchronization solutions in the world. An established component of Starfish Managed Mobility (TM) solutions, the TrueSync infrastructure is designed to provide end-to-end synchronization for:

DATA SHEETS

- Product Overview

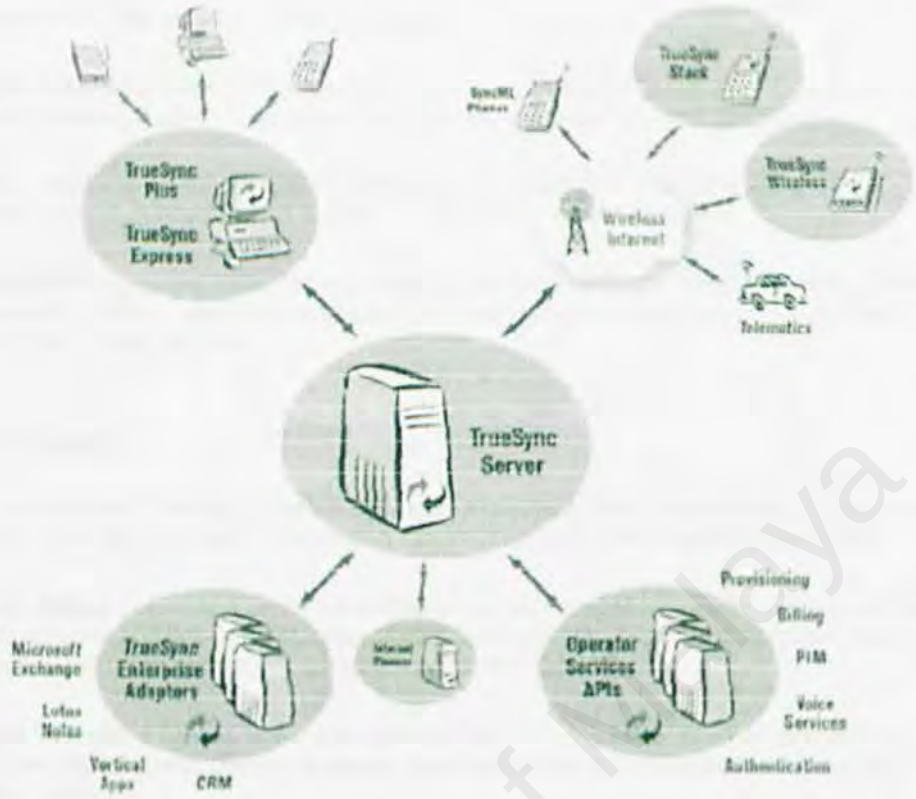
- Wireless Data Services
- Enterprise Applications
- Mobile Devices
- Corporate Infrastructures

The TrueSync technology platform is a comprehensive set of wireless and wireline synchronization products and solutions.

TrueSync solutions support a wide universe of devices, desktop applications, enterprise server applications, and web services. The TrueSync technology platform's multi-point architecture is highly scalable and designed for enterprise, wireless operator and device manufacturer deployments.

Click on the individual elements in this diagram for detailed information, or scroll further for brief overviews.

University of Malaya



Back to Top

Server Products and Solutions

TrueSync Server Products and Solutions provide multi-tier over-the-air and wireline synchronization between a heterogenous mix of devices, applications and services.

The Java-based architecture easily integrates with existing software applications and supports a wide variety of platforms, databases, applications, protocols and transports for broad interoperability.

TrueSync Server Products include:

- **TrueSync Server**
- **TrueSync Enterprise Adaptors**
- **TrueSync Server APIs**
- **Internet Planner**
- **TrueSync Thin Client**

TrueSync Server Solutions include:

- **TrueSync Mobile Enterprise Solution**
- **TrueSync Wireless Operator Solution**

Desktop Products

Software and development tools enabling synchronization among mobile devices, desktop

University of Malaya

applications and web services, using TrueSync technology.

TrueSync Plus synchronization software providing one-click patented multi-point data sync between Windows platforms and popular devices, applications and services.

TrueSync Express synchronization software providing fast and easy sync of Outlook/Exchange contents with a user's mobile phone address book.

TrueSync SDK software tools for developing customized accessors, providing desktop sync for connected devices, Windows applications, and network-based services, for both standard and proprietary data sources.

Device Products

SyncML-conformant device software products provide device manufacturers the ability to implement standardized synchronization capabilities for PDAs, handsets and PCs.

TrueSync Stack optimized embedded device software enables quick and easy integration of the latest open-standard SyncML capabilities into devices, facilitating synchronization with TrueSync servers and any other SyncML-compliant servers.

TrueSync Wireless device software applications provide after-market SyncML capabilities for over-the-air synchronization between handheld device applications and backend application servers.

TrueSync Licensing

To find out how to license TrueSync technology for use with devices, applications and servers, contact Starfish Sales or call 888 961 9696.

---

http://www.airtuit.com/

- BLUEMOON MICRO EDITION
- BLUEMOON STUDIO
- BLUEMOON APPLICATION SERVER
- BLUEMOON INBOX
- DOWNLOADS

BLUEMOON

PRODUCTS

BlueMoon M.E. (Micro Edition) for Windows CE devices is an innovative solution for wireless data applications in areas with unpredictable service. Built on Java technology, utilizing a Jeode platform, a mobile user can now work effective in either online or offline modes.

University of Malaya

Click here for a larger screenshot.

Click here for a larger screenshot.

University of Malaya

Appendix - User Manual

1.1 Introduction

This manual is divided mainly into two parts, which are Client software and Server Software.

1.2 Client Software

1.2.1 Splash Screen and entering the IIS

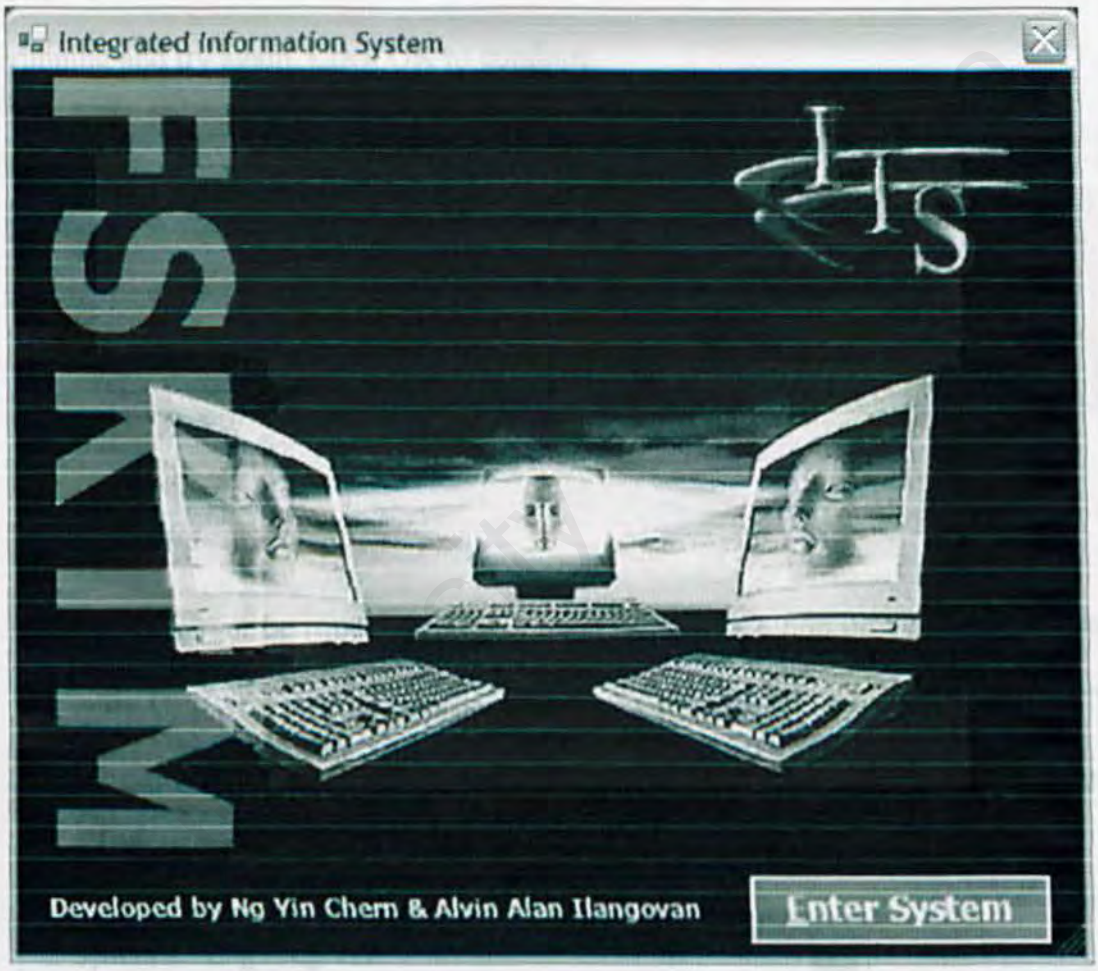


Figure 1.1 : Splash Screen

This is the splash screen. It would be the first window to pop up when you click the icon on the desktop that will allow you to open the IIS. Simply click the Enter System button to proceed into the IIS.

Figure 1.2 : Connection Message

Should the server not be turned on or is not active the window like figure B would pop-up. This means that the server is not running . Make sure that the server is running and try again. Click OK to continue.

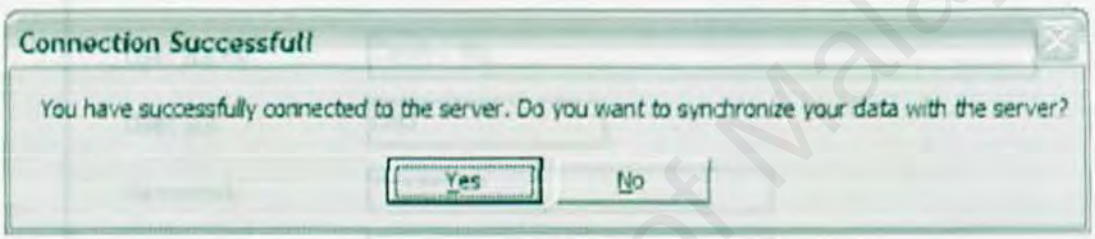


Figure 1.3

Should the server be running you will be greeted with a pop up box which is shown in Figure C. If you wish to synchronize your data with the server click on the 'Yes' button and on the 'No' button if you do not wish to synchronize your data at that present time.

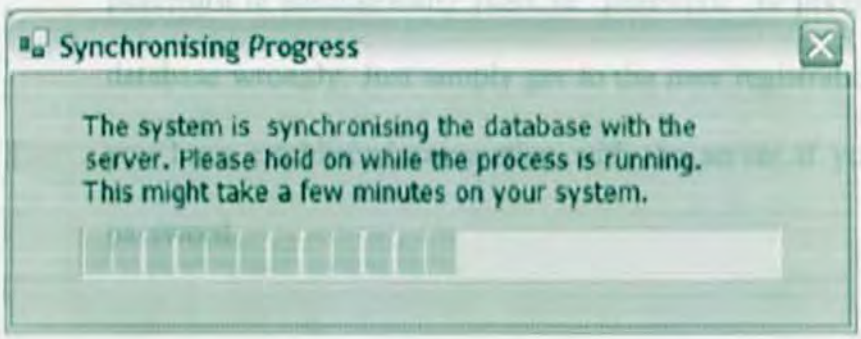


Figure 1.4: Synchronization Dialog

University of Malaya

Should you click 'Yes" you will notice the sync dialog loading up. This Status bar shows the status of the system as it synchronizes with the server and other clients. This dialog will also be seen every time you click on the 'Synchronize with Server' button located in the main menu of the system

1.2.3 User Registration

1. The first time you run the software, it will prompt you to enter your user information. Just enter all the information as follows.

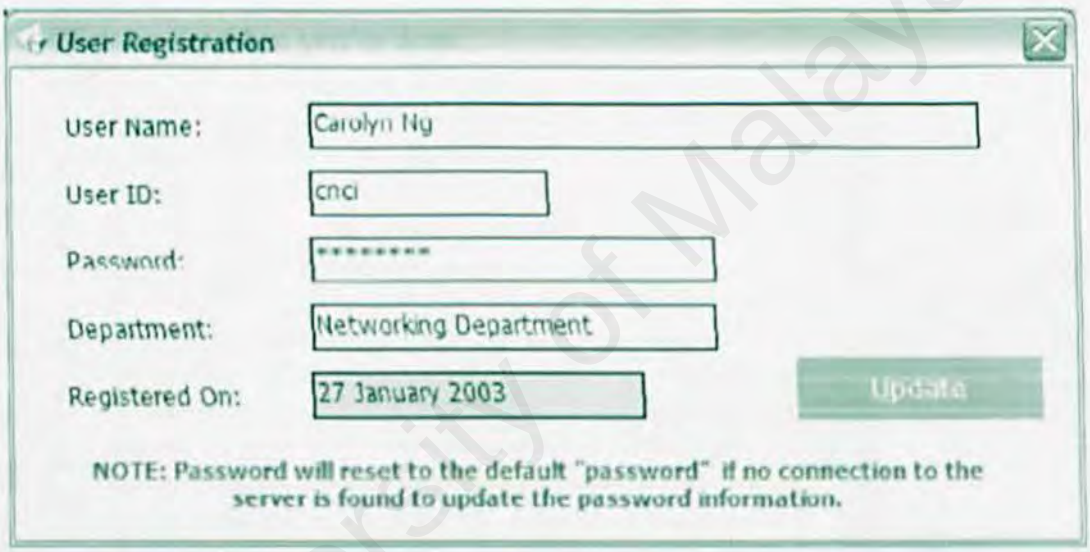


Figure 1.5: User Registration

2. After entering all the required data, just click on "Update" to log into the software. Take note that if the default server setting is not correct, the password is automatically reset to "password" to prevent it updating of the database wrongly. Just simply get to the user registration screen again when you have established connection with the server if you want to change to password.

University of Malaya

1.2.3 Server Settings

Figure 1.6: Server Settings

1. This allows the users to point the software to connect to the IIS Server.
2. Enter the two required information which are the HostName and Port.
3. Click on Set when you're done.

University of Malaya

1.2.4 The Main Window

Once you have entered the system , you will see the main window. Figure C represents what you will see when you enter the system.

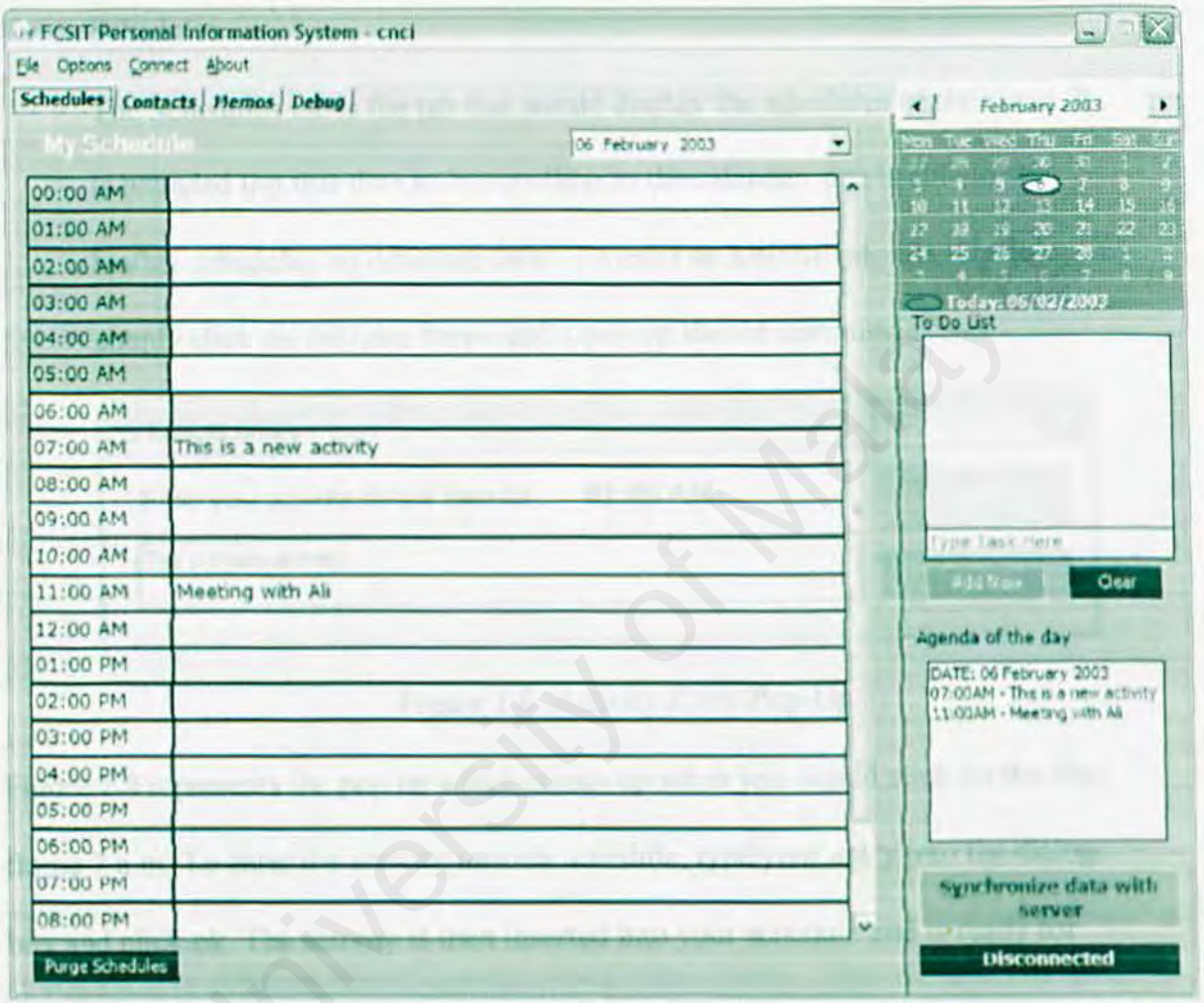


Figure 1.7 : Main Menu

It consists of four tabs on the menu which are schedules, contacts, memo and debug. The debug option is only put for the viva and will not be present should the IIS be commercialized. The default screen would be that of the schedules tab. On the right of the screen you will notice three 'box columns'. They are the calendar, the 'to do list' and the 'agenda of the day' column. Below these boxes is a button (Synchronize Data With Server) . When this button is clicked it will start to synchronize all the

entries the client has made with the server. Below this button is the status box which shows if the system is connected to the server or not.

The three tabs will now be explained in more detail:

I) Schedules

The Schedules tab is the tab that would display the schedules of the client. It is bounded top that date so by scrolling to the calendar, one is able to see his/her schedules on different dates. To enter an activity into the schedule simply click on the time frame and a pop up should commence:

Figure 1.8 :Activity Entry Pop-Up

Figure 1.8 represents the pop up which comes up when you would click on the time frame 7 a.m. To enter the activity into the schedule, type your entry into the dialog box and click ok. The activity is then inserted into your schedule and is ready for synchronization.

University of Malaya

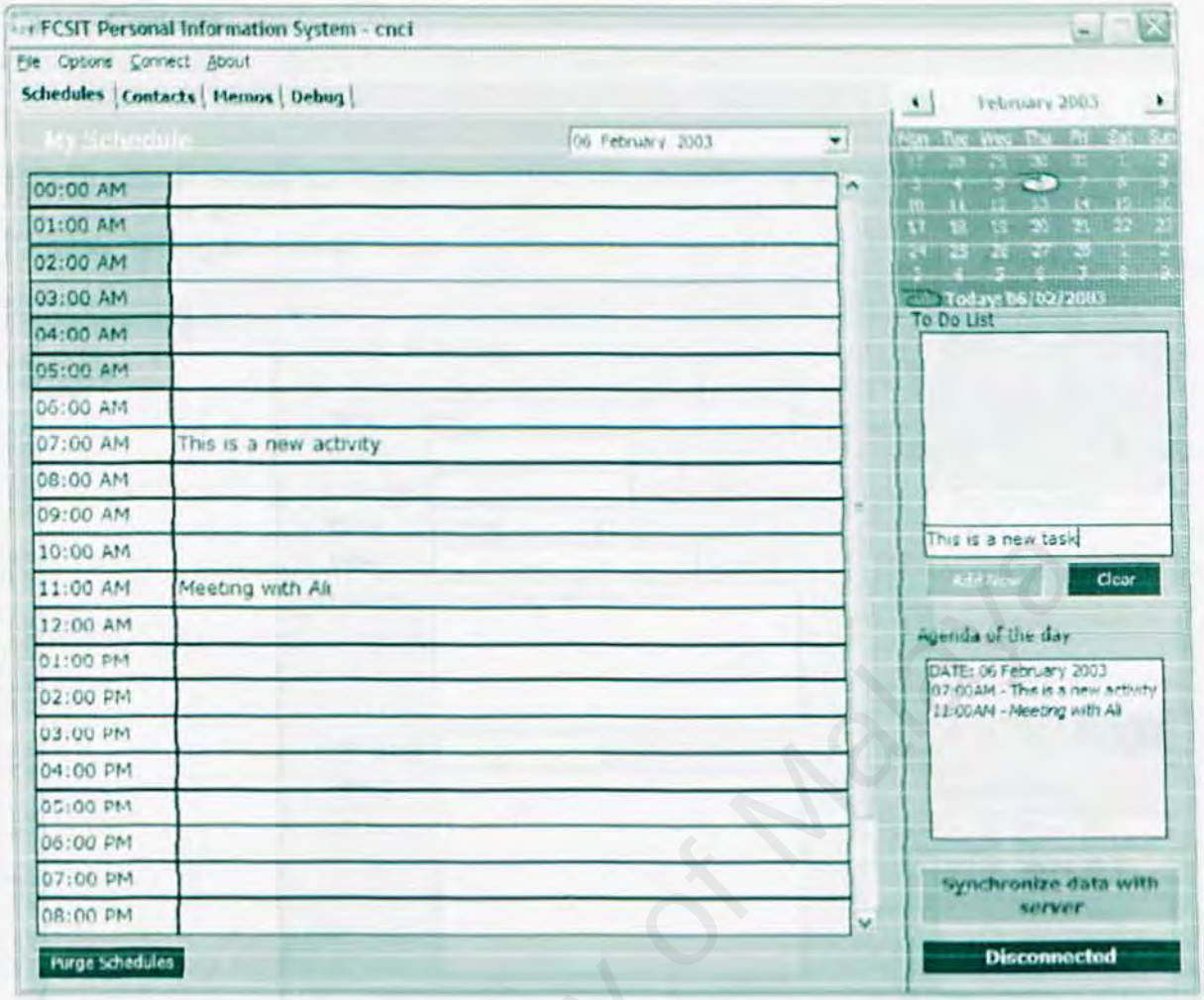


Figure 1.9: Activity Entry Successful

Figure 1.9 shows that the activity entered in Figure 1.8 has been entered into the schedule at 7 a.m.

If you want to delete an entry , just click on the time frame , erase the data and click ok and time frame will be empty again.

It is worth while to mention the purge schedules in the bottom left corner of the window. When this button is clicked it will clear all the schedules that are in the time frames thus making it simpler to erase all entries on that day if it is nessacary.

University of Malawi

II) Contacts

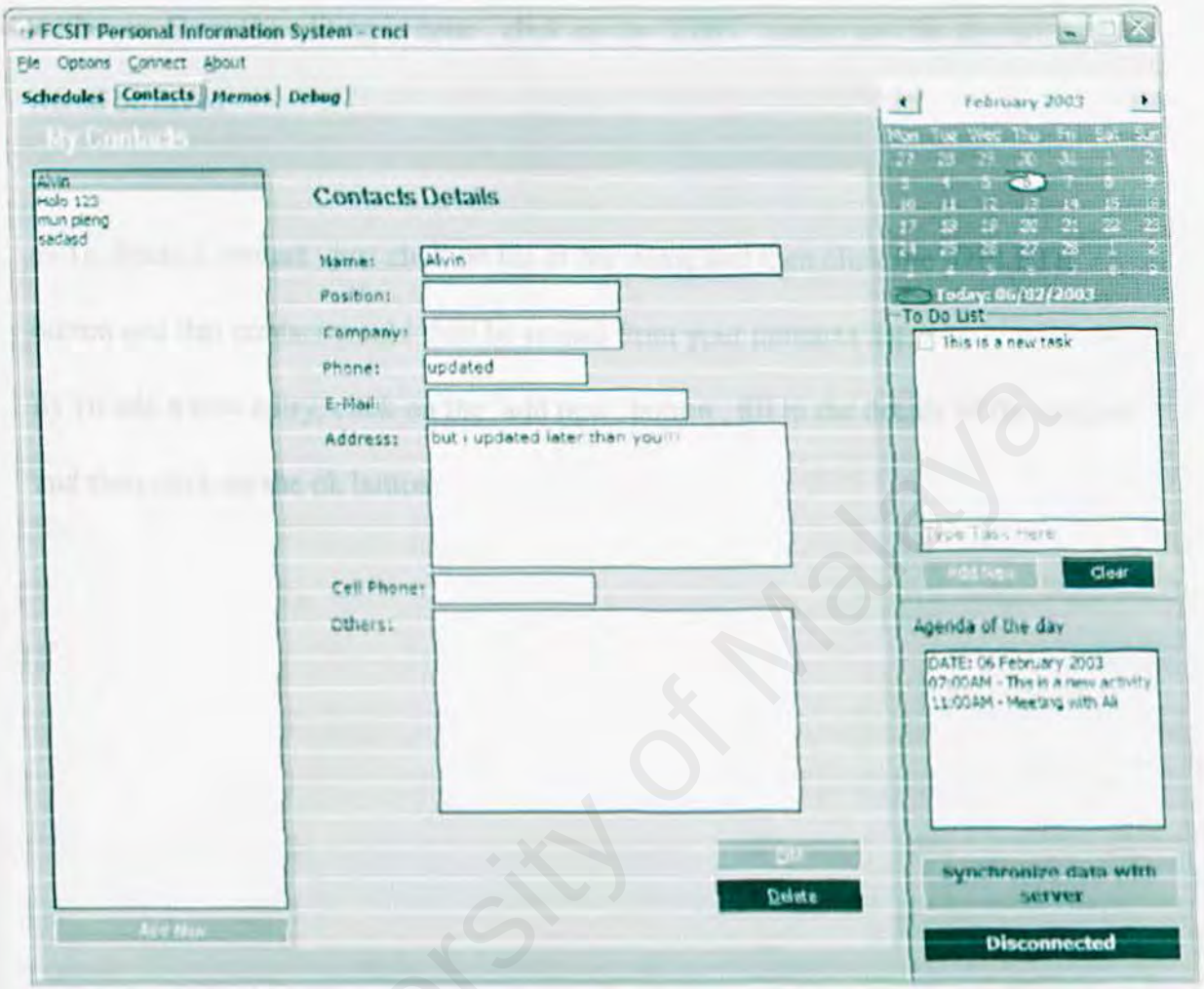


Figure 2.0 : Contacts Window

a) The next tab would be the Contacts tab. When this tab is clicked you will be able to see a screen similar to the one is Figure 2.0. In this example we have 4 entries already in our address book. The list of contacts are located in the right hand corner in a listbox. When we want to view the details of a contact we simply click on his name and the details that were entered about him/her would appear in the dialog boxes. Among the details that we can view are the contacts name, company, address, cell phone number, telephone number, e mail and the position of him or her at the company.

University of Malaya

b) To edit the details of the contact, click on his/her name and edit the details in the textboxes. Once the editing is done , click on the 'EDIT' button and the alteration would be saved.

c) To delete a contact , just click on his or her name and then click the 'DELETE' button and that contact would then be erased from your contacts list.

d) To add a new entry, click on the 'add new' button , fill in the details of the contact and then click on the ok button.

University of Malaya

III Memos

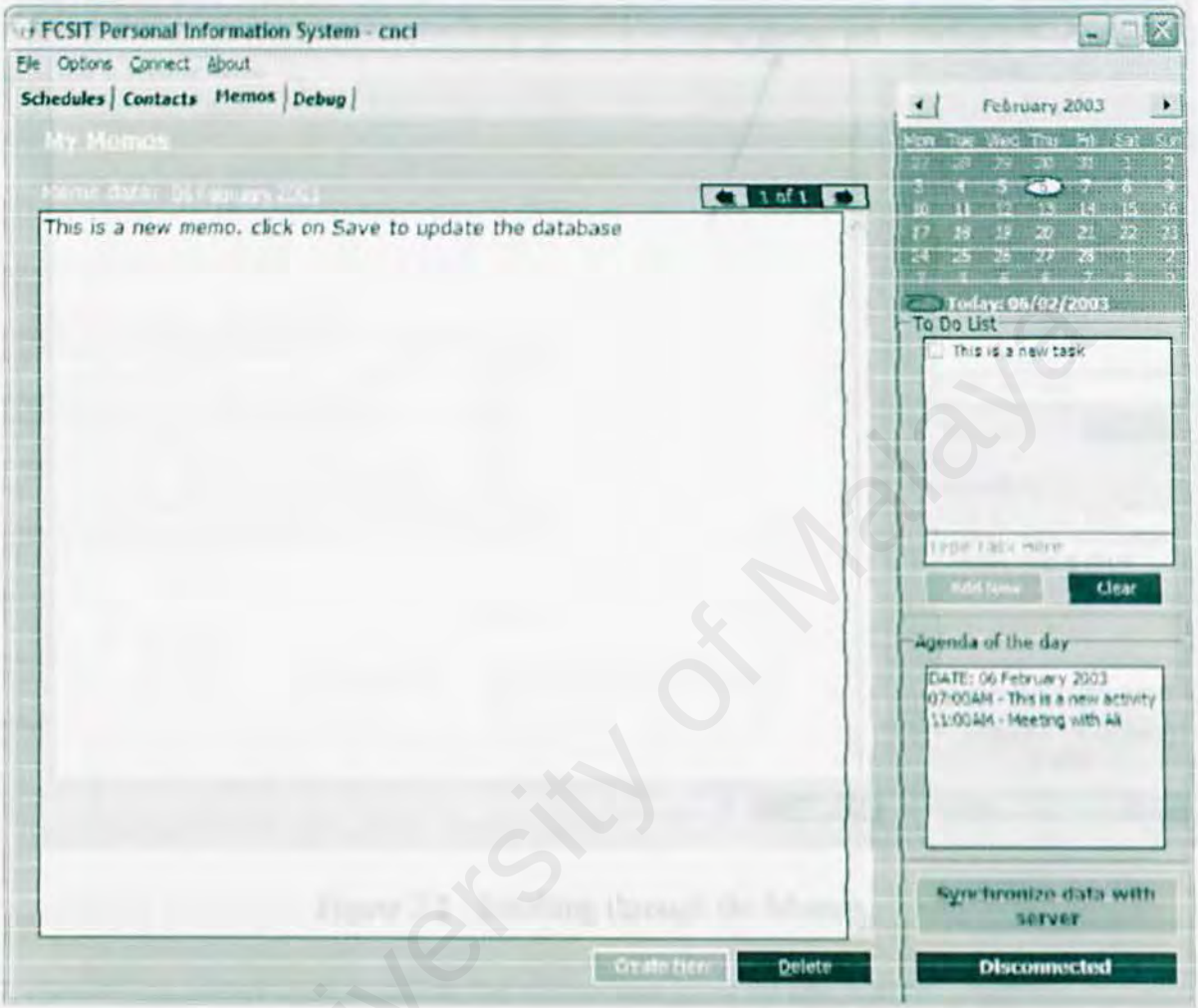


Figure 2.1:Memos Window

a) Figure 2.1 shows the display of the application when the memos tab is clicked. This tab allows the user to input memos into his/her database which could later be synchronized with the server and other clients. Several memos can be entered and each memo is numbered and the numbering is shown at the top right corner of the tab section of the window.

University of Malaya

i)Browsing through the memos

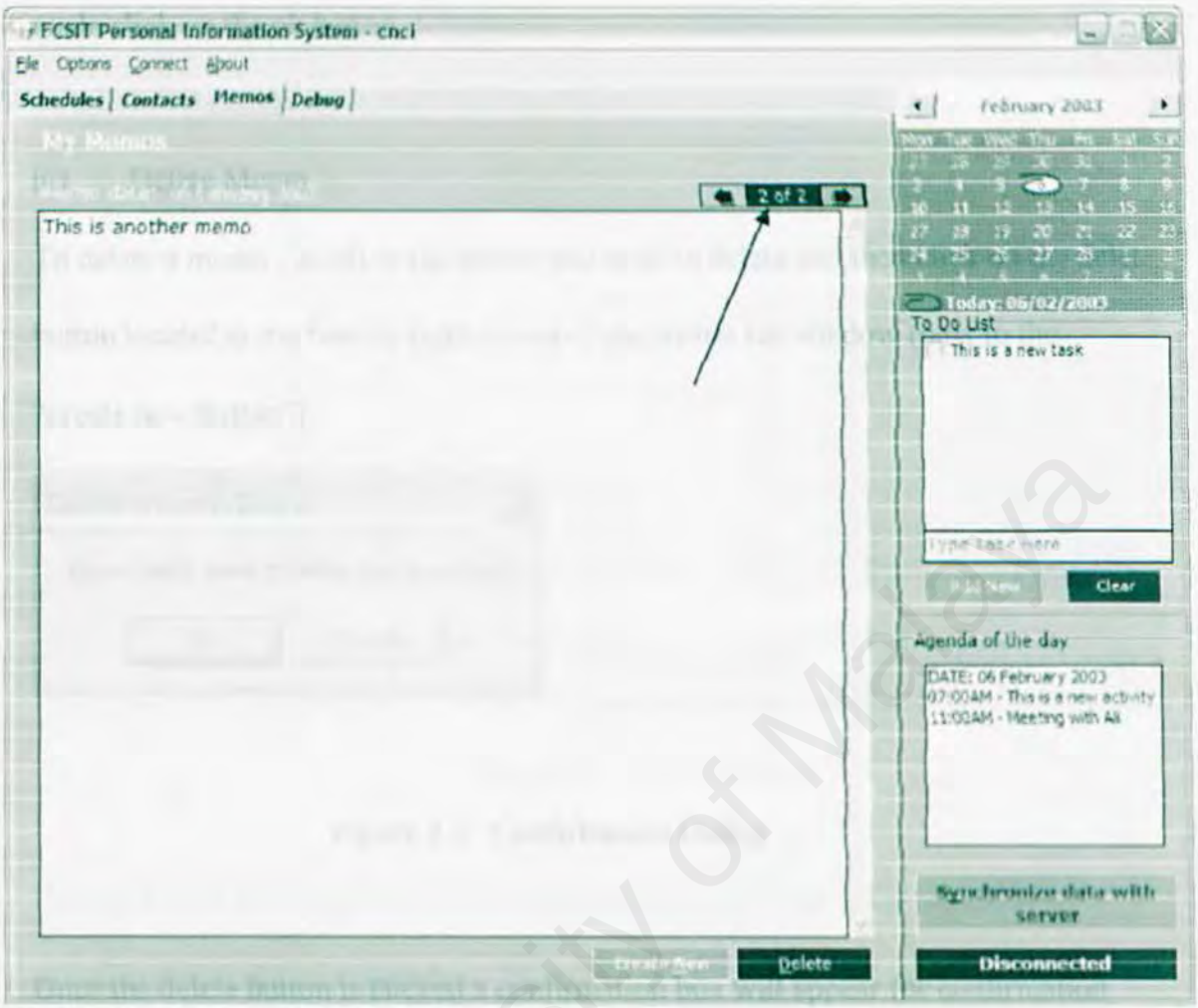


Figure 2.2 : Scrolling through the Memos

To browse through the memos you can use the scrolling bar at the top right corner of the memos tab which also shows the number of memos in your database as shown with the arrow in figure 2.2.

ii) Adding memos

To add a memo click on the create new button on the bottom right corner of the memo tab window and then enter your memo into the dialog list box (in Figure H

the message that was written in this space was "This is another memo") Once entry is made click on the ok button.

iii) Delete Memo

To delete a memo , scroll to the memo you wish to delete and then click on the delete button located at the bottom right corner of the memo tab window (next to the "create new button").

Figure 2.3 : Confirmation Dialog

Once the delete button is clicked a confirmation box will appear for confirmation regarding deleting the memo. If you want to delete your memo click the 'OK' button , and if you do not wish to delete the memo then click on the 'Cancel' button.

University of Malaya

1.3 Server Software

1.3.3 User Management

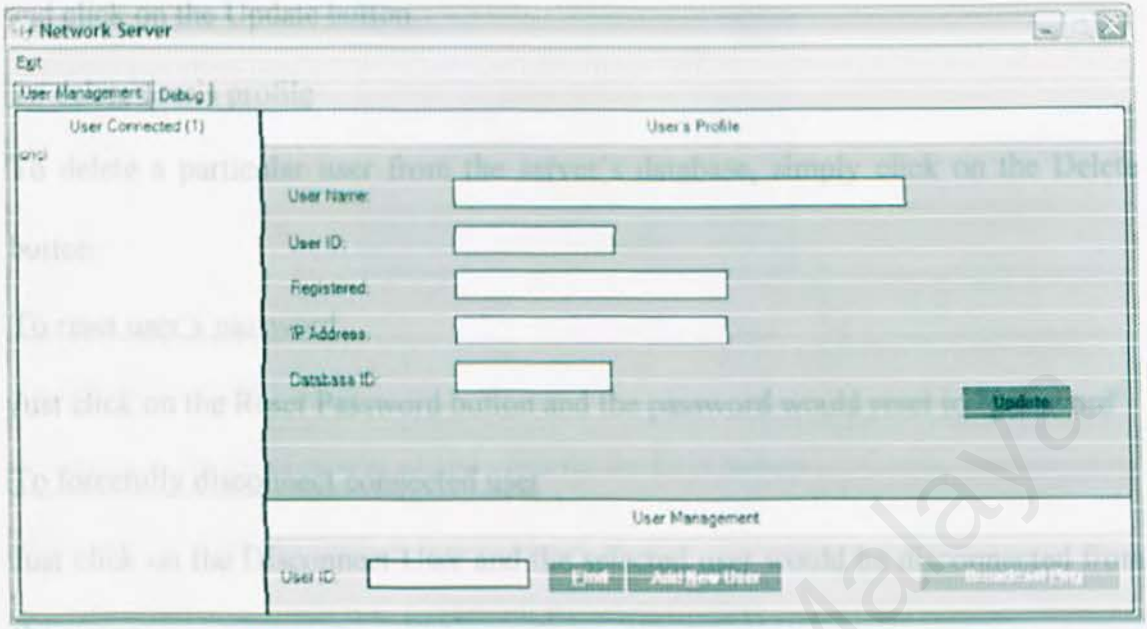


Figure 2.4: Server Software

The IIS Server software has a very simple but straight to the point interface.

To view connected user's profile

1. Simply click on the user id from the connected user list and the information would be shown in the User's Profile section.

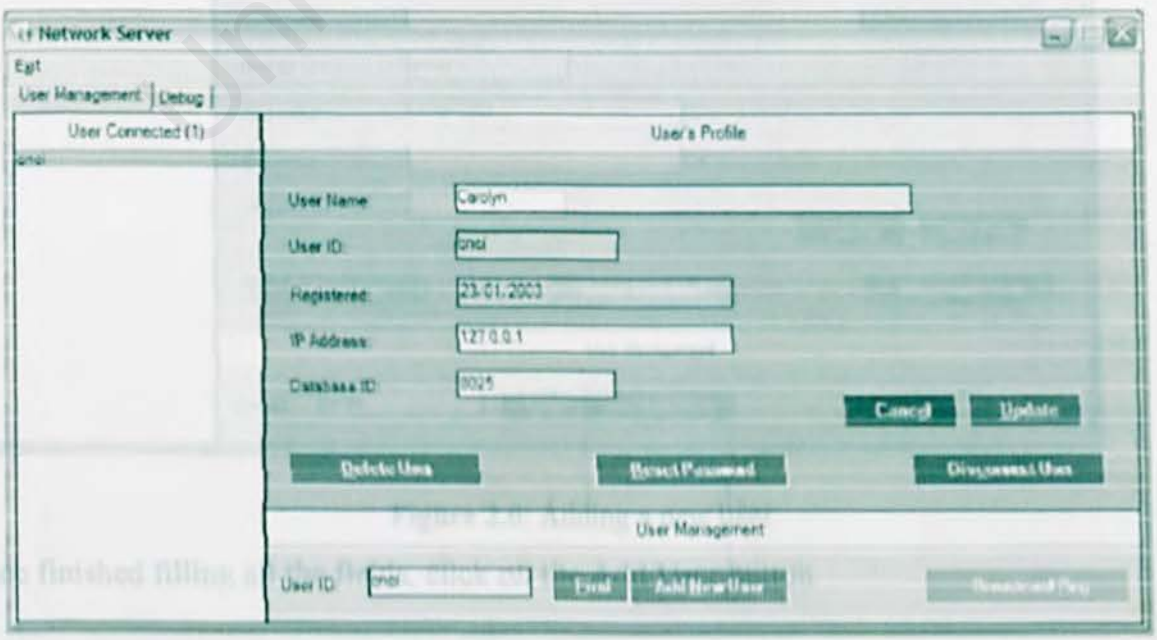


Figure 2.5: Viewing connected user's profile

University of Malaya

To update user's profile

Simply make all the appropriate update to the fields within the user's profile section and click on the Update button.

To delete user's profile

To delete a particular user from the server's database, simply click on the Delete button

To reset user's password

Just click on the Reset Password button and the password would reset to "password"

To forcefully disconnect connected user

Just click on the Disconnect User and the selected user would be disconnected from the server.

To add new user

Just type in the new user id into the user id field and click in the Add New User button.

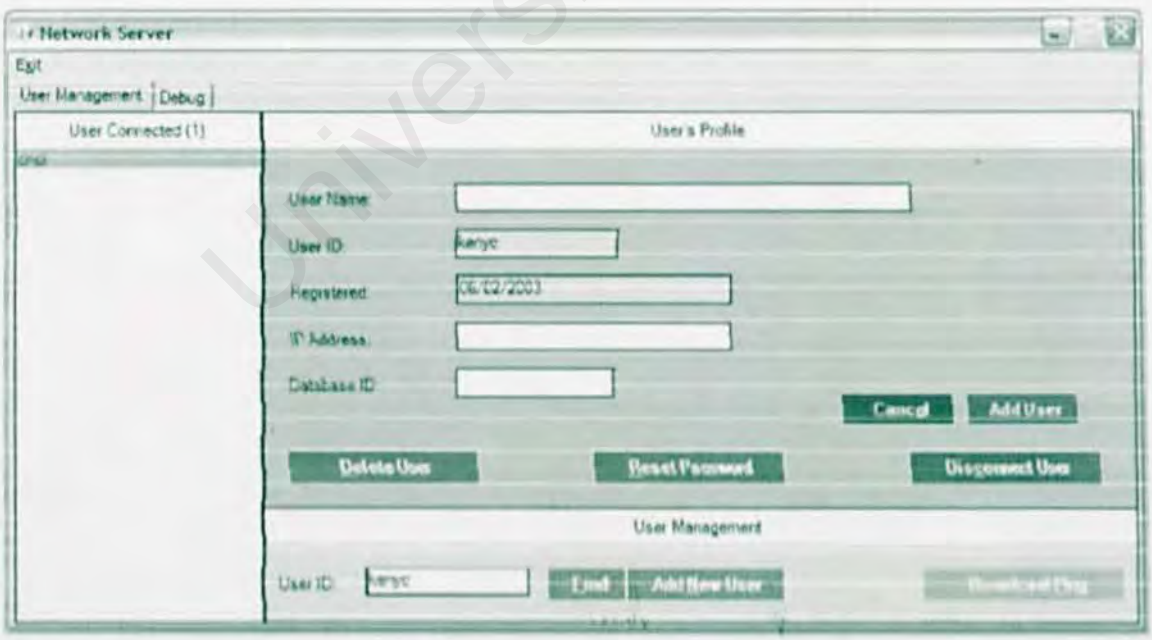
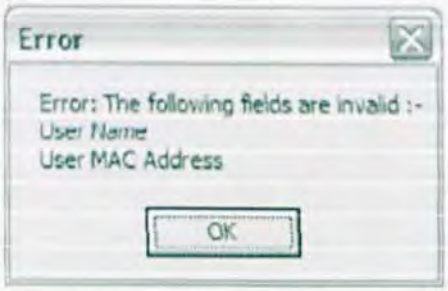


Figure 2.6: Adding a new user

Once finished filling all the fields, click on the Add User button.

University of Malaya

In software would check the entered data for invalid entries. In the case of errors, a message box would appear stating the fields with the errors.



2.7: Add New User Error

To find an existing user id

Just type in the existing user id and click on the Find button.

If the user id does not exist, it'll prompt out an error message.

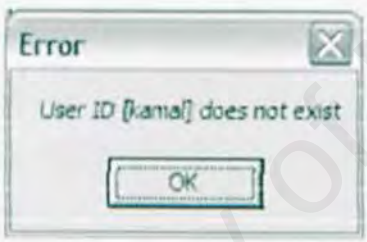


Figure 2.8: Find User Error

University of Malaya